

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA DE TELECOMUNICACIONES EN SONIDO E IMAGEN



PROYECTO FIN DE CARRERA

**Desarrollo de una Aplicación de Subtitulado para el Gestor de Contenido
Multimedia ARCAMM de la Universidad Carlos III de Madrid**

AUTOR: SONIA MARTÍN CARMONA

TUTOR: VICENTE PALACIOS MADRID

Diciembre, 2010

Agradecimientos

Después de terminar este proyecto, hay varias personas a las que me gustaría agradecer no sólo su ayuda en la elaboración del mismo, sino su amistad y apoyo durante este tiempo.

A los primeros que quiero dar las gracias son mis padres, por quererme, apoyarme, ayudarme en todo lo que han podido... por todo, muchas gracias. También agradecer a toda mi familia: mi hermano, tíos, primos y abuelos. Porque me siento muy orgullosa de contar con todos ellos.

En segundo lugar, quiero dar las gracias a Juan. Por su cariño, su confianza, sus ánimos, su alegría, su paciencia... en definitiva, por compartir su vida conmigo.

Los siguientes a los que quiero agradecer son Marta y David. Primero, por toda su ayuda, incluyendo en este proyecto. Y segundo, y más importante, por su amistad y los buenos momentos que me han ofrecido desde que los conozco.

También quiero dar las gracias a mi tutor, Vicente, por sus consejos y sus recomendaciones durante estos meses.

No me olvido del resto de mis amigos, que me han ayudado y animado constantemente y de los que tengo momentos inolvidables. Entre ellos, además, me gustaría destacar a Juanma, cuya ayuda ha sido de gran importancia en este proyecto.

Gracias también al área de audiovisuales y a aquellas personas que han aportado su granito de arena.

Por último, quiero dar las gracias a una persona especial, Ana. Por enseñarme que hay que luchar y ser valiente hasta el final.

Resumen

Las recomendaciones incluidas en las pautas WCAG, creadas por el *World Wide Web Consortium*, tienen como finalidad garantizar la accesibilidad de los contenidos en la Web, incluyendo el caso particular de los contenidos audiovisuales. Para cumplir estas recomendaciones con los contenidos audiovisuales se utiliza, entre otras técnicas, el subtitulado.

Actualmente, existen diversas herramientas de subtitulado, tanto accesibles vía web como ejecutables en local. No obstante, es complicado encontrar una herramienta que permita realizar una subtitulación on-line, con corrección de errores e independiente del propietario del vídeo a subtitular.

El objetivo de este proyecto es desarrollar una aplicación que, además de cubrir los requisitos anteriores, se integre con el Gestor *ARCA Media Manager* de la Universidad Carlos III de Madrid. De este modo, la aplicación soportará la gestión completa de subtítulos, esto es: creación, edición, eliminación y reproducción, en conjunción con el contenido audiovisual que complementan, compartido por el Gestor ARCAMM.

Índice general

CAPÍTULO 1. INTRODUCCIÓN	1
1.1. Motivación	1
1.2. Objetivos	2
1.3. Estructura del proyecto	4
 CAPÍTULO 2. ESTADO DEL ARTE	 5
2.1. Gestión de Contenido Multimedia	6
2.1.1. El contenido multimedia y audiovisual.....	6
2.1.2. Gestores de Contenido Multimedia	8
2.1.3. Federadores de Contenido Multimedia	14
2.2. Accesibilidad Audiovisual	21
2.2.1. Accesibilidad Web	21
2.2.2. Accesibilidad a los contenidos audiovisuales	23
2.3. La subtitulación	30
2.3.1. Tipos de subtitulación	32
2.3.2. Lenguajes de subtitulación	35
2.3.3. Formatos de subtítulos.....	36
2.4. Herramientas de subtitulado.....	42
2.4.1. Herramientas en PC.....	43
2.4.2. Herramientas on-line	49
2.4.3. Comparación de herramientas	51
2.5 UML (Unified Modeling Language).....	53
2.5.1. Diagrama de casos de uso	53
2.5.2. Diagrama de despliegue	54
2.5.3. Diagrama de clases	55
2.5.4. Diagramas de estados.....	57
2.5.5. Diagrama de actividades	57

CAPÍTULO 3. HERRAMIENTAS PARA LA ELABORACIÓN DEL PROYECTO	59
3.1. Infraestructura del Servidor	59
3.1.1. Servidor Web	60
3.1.2. Sistema de Gestión de Bases de Datos	62
3.2. Entorno de trabajo	64
3.2.1. Adobe Flash y ActionScript	64
3.2.2. PHP (Hipertext Preprocessor)	67
3.2.3. JavaScript	69
3.2.4. XML (eXtensible Markup Language)	71
3.2.5. EditPlus	74
3.3. Librerías	76
CAPÍTULO 4. DESARROLLO DEL PROYECTO	80
4.1. Fase inicial	81
4.1.1. Proceso de desarrollo	82
4.1.2. Especificación inicial de requisitos	83
4.1.3. Diagrama de casos de uso inicial	85
4.2. Fase de análisis	86
4.2.1. Diagrama de casos de uso en la fase de análisis	86
4.2.2. Especificación de requisitos en la Fase de Análisis	90
4.2.3. Diagrama de clases del dominio	101
4.2.4. Integración en el Gestor ARCAMM	103
4.2.5. Comunicación con la base de datos de ARCAMM	106
4.2.6. Archivo de subtítulos	108
4.3. Fase de diseño arquitectónico	109
4.3.1. Infraestructura Software	110
4.3.2. Infraestructura Hardware	112
4.3.3. Reproductor	113
4.3.4. Editor de subtítulos	115
4.3.5. Archivo XML de subtítulos	116

4.4. Fase de diseño detallado	118
4.4.1. Implementación del reproductor	118
4.4.2. Descripción de la aplicación de subtitulado	128
4.5. Resumen del proyecto	135
4.5.1. Planificación final	135
4.5.2. Resumen de costes	137
4.5.3. Estado del proyecto	138
 CAPÍTULO 5. CONCLUSIONES	 140
5.1. Futuras líneas de desarrollo	142
 CAPÍTULO 6. BIBLIOGRAFÍA	 144
 CAPÍTULO 7. GLOSARIO	 153

Lista de Figuras

Figura 2.1. Clasificación de contenidos multimedia	7
Figura 2.2. Ejemplo de PuMuKIT en la Universidad de Vigo	10
Figura 2.3. Ejemplo de MediaMosa	11
Figura 2.4. Ejemplo de DSpace en la librería digital de Texas	12
Figura 2.5. Portal de ARCAMM en la Universidad Carlos III de Madrid	14
Figura 2.6. Comunicación entre Federador y Gestores	15
Figura 2.7. Captura de pantalla de Steeple	16
Figura 2.8. Captura de pantalla de iTunes U	17
Figura 2.9. Captura de pantalla de agrega	19
Figura 2.10. Logo de ARCA.....	20
Figura 2.11. Captura de pantalla de ARCA	20
Figura 2.12. Modelo de requerimientos para la accesibilidad audiovisual.....	24
Figura 2.13. Ejemplo de subtítulo.....	27
Figura 2.14. Error de códec	30
Figura 2.15. Subtítulos DVB (izquierda) y Subtítulos de Teletexto (derecha)	31
Figura 2.16. Ejemplo de colores en subtítulos	33
Figura 2.17. Máquina de estenotipia.....	34
Figura 2.18. Captura de pantalla de la herramienta Subtitle Workshop	44
Figura 2.19. Captura de pantalla de la herramienta Aegisub.....	46
Figura 2.20. Captura de pantalla de la herramienta MAGpie	47
Figura 2.21. Captura de pantalla de la herramienta Hi-Caption	48
Figura 2.22. Captura de pantalla de la herramienta Overstream	50
Figura 2.23. Ejemplo de casos de uso.....	54
Figura 2.24. Ejemplo de diagrama de despliegue	55
Figura 2.25. Representación gráfica de un paquete	56
Figura 2.26. Representación gráfica de una clase	56
Figura 2.27. Ejemplo diagrama de estados	57
Figura 2.28. Ejemplo diagrama de actividades.....	58

Figura 3.1. Arquitectura cliente/servidor	61
Figura 3.3. Logo de MySQL	63
Figura 3.4. Logo de <i>Adobe Flash</i>	65
Figura 3.5. Logo de PHP	67
Figura 3.6. Captura de pantalla de la herramienta EditPlus.....	76
Figura 4.1. Casos de uso inicial	85
Figura 4.2. Casos de Uso en la fase de análisis.....	87
Figura 4.3. Diagrama de clases del dominio.....	102
Figura 4.4. Validación de usuario en ARCAMM.....	104
Figura 4.5. Elección de roles en ARCAMM	104
Figura 4.6. Página ítem de ARCAMM para el rol Manager.....	105
Figura 4.7. Esquema del archivo de subtítulos.....	108
Figura 4.8. Capas de la arquitectura	110
Figura 4.9. Arquitectura Software	111
Figura 4.10. Arquitectura Hardware.....	112
Figura 4.11. Reproductor-Componentes de Flash	114
Figura 4.12. Arquitectura del Editor de subtítulos	115
Figura 4.13. Flujo del funcionamiento del reproductor	119
Figura 4.14. Ejemplo de estados para el botón Pause	121
Figura 4.15. Diagrama de estados del botón Play/Pause.....	122
Figura 4.16. Diagrama de estados del botón Stop	123
Figura 4.17. Diagrama de estados del Botón Mute.....	123
Figura 4.18. Diagrama de estados del Botón Caption	124
Figura 4.19. Barra de Volumen.....	124
Figura 4.20. Barra de reproducción/carga de buffer	125
Figura 4.21. Etiqueta de tiempos	125
Figura 4.22. Diagrama de flujo del reproductor	128
Figura 4.23. Selección del rol “subtitulador”	128
Figura 4.24. Página ítem del rol “subtitulador”	129
Figura 4.25. Página de la aplicación de subtítulos.....	130

Figura 4.26. Iconos de izquierda a derecha: Crear subtítulos, Crear archivo XML, Borrar archivo XML	131
Figura 4.27. Ejemplo de subtítulo.....	131
Figura 4.28. Botón “Añadir subtítulo” y marca de tiempo.....	132
Figura 4.29. Guardar subtítulos en archivo	133
Figura 4.30. Borrar archivo de subtítulos	133
Figura 4.31. Ventana para importar archivo SubRip	134
Figura 4.32. Iconos de izquierda a derecha: Volver a página ítem, Importar, Cerrar sesión	135
Figura 4.33. Diagrama de Gantt	136

Lista de Tablas

Tabla 2.1. Características de las WCAG 1.0 y las WCAG 2.0.....	24
Tabla 2.2. Comparativa de herramientas de subtitulación	52
Tabla 4.1. CU-001	88
Tabla 4.2. CU-002	88
Tabla 4.3. CU-003	88
Tabla 4.4. CU-004	89
Tabla 4.5. CU-005	89
Tabla 4.6. CU-006	89
Tabla 4.7. CU-007	89
Tabla 4.8. CU-008	90
Tabla 4.9. CU-009	90
Tabla 4.10. CU-010	90
Tabla 4.11. RU-001	91
Tabla 4.12. RU-002	91
Tabla 4.13. RU-003	91
Tabla 4.14. RU-004	92
Tabla 4.15. RU-005	92
Tabla 4.16. RU-006	92
Tabla 4.17. RU-007	92
Tabla 4.18. RU-008	93
Tabla 4.19. RU-009	93
Tabla 4.20. RU-010	93
Tabla 4.21. RU-011	93
Tabla 4.22. RU-012	94
Tabla 4.23. RU-013	94
Tabla 4.24. RU-014	94
Tabla 4.25. RU-015	94
Tabla 4.26. RU-016	95

Tabla 4.27. RU-017	95
Tabla 4.28. RU-018	95
Tabla 4.29. RU-019	95
Tabla 4.30. RU-020	96
Tabla 4.31. RU-021	96
Tabla 4.32. RU-022	96
Tabla 4.33. RU-023	97
Tabla 4.34. RU-024	97
Tabla 4.35. RU-025	97
Tabla 4.36. RU-026	97
Tabla 4.37. RU-027	98
Tabla 4.38. RU-028	98
Tabla 4.39. RU-029	98
Tabla 4.40. RU-030	98
Tabla 4.41. RU-031	99
Tabla 4.42. RU-032	99
Tabla 4.43. RU-033	99
Tabla 4.44. RU-034	99
Tabla 4.45. RU-035	100
Tabla 4.46. RU-036	100
Tabla 4.47. RU-037	100
Tabla 4.48. RU-038	100
Tabla 4.49. RU-039	101
Tabla 4.50. RU-040	101
Tabla 4.51. Tabla Ítem de la base de datos	106
Tabla 4.52. Resumen de días y horas por fase	137
Tabla 4.53. Coste de material	138
Tabla 4.54. Coste total del proyecto	138

Lista de Ejemplos

Ejemplo 2.1. Subtítulos con el formato SubRip	39
Ejemplo 2.2. Etiqueta para letra cursiva en el formato de subtítulos SubRip	40
Ejemplo 2.3. Etiqueta para letra cursiva en el formato de subtítulos SSA.....	40
Ejemplo 2.4. Ejemplo de texto en cursiva correcto	41
Ejemplo 2.5. Ejemplo de texto en cursiva incorrecto	41
Ejemplo 2.6. Subtítulo con efectos.....	41
Ejemplo 2.7. Ejemplo de subtítulos con etiqueta para la fuente y el estilo cursiva del texto.....	42
Ejemplo 4.1. DTD del archivo de subtítulos	116
Ejemplo 4.2. Ejemplo de código del archivo de subtítulos	117

CAPÍTULO 1. INTRODUCCIÓN

1.1. Motivación

Actualmente, Internet se emplea en muchas facetas de nuestra vida diaria ya que existen aplicaciones de todo tipo, como correo, redes sociales, juegos, etc. pero, sobre todo, se ha extendido el uso de aplicaciones que manejan contenidos audiovisuales, como vídeos o audio.

Debido a que la Web sigue aumentando cada día en número y tipo de usuarios, las aplicaciones de Internet y sus contenidos deben poder ser accesibles para cualquier persona. Sin embargo, encontramos muchos sitios web que presentan dificultades al querer acceder a ellos y con contenidos no accesibles para todos los usuarios.

El problema de la accesibilidad existe en muchos medios y para intentar solucionarlo se han estudiado muchas técnicas, las cuales pueden aplicarse también a la accesibilidad en las páginas web.

Las recomendaciones incluidas en las pautas WCAG (Pautas de Accesibilidad al Contenido en la Web), creadas por el *World Wide Web Consortium*, tienen como finalidad garantizar la accesibilidad de los contenidos en la Web, incluyendo el caso particular de los contenidos audiovisuales. Para cumplir estas recomendaciones con los contenidos audiovisuales se utiliza, principalmente, las técnicas de subtitulado, audiodescripción y lengua de signos.

En el ámbito del subtitulado existen diversas herramientas que, en su mayoría, deben ser instaladas en local. Actualmente están surgiendo otras accesibles a través de Internet.

Las herramientas locales, en general, suelen ser editores de subtítulos que permiten dar formato al texto, realizar varias operaciones con los tiempos y controlar posibles

errores que puedan surgir durante la subtitulación. Desafortunadamente, requieren que los vídeos se encuentren en el equipo y no siempre son multiplataforma.

Las herramientas on-line son independientes del sistema operativo y proporcionan la posibilidad de subtitular vídeos que no son propios, obtenidos de gestores de vídeo, como YouTube. Para ello, permiten introducir el texto y los tiempos para cada subtítulo. Sus principales deficiencias es que no proporcionan la posibilidad de editar el formato y color del texto de los subtítulos ni comprueban si existe algún tipo de error en la subtitulación.

En resumen, se puede decir que es complicado encontrar una herramienta que permita realizar una subtitulación on-line, con gestión de formato y control de errores, independientemente del propietario del vídeo que se desee subtitular.

1.2. Objetivos

Los objetivos de este proyecto han sido establecidos a partir de las necesidades comunicadas por el Área de Audiovisuales del Servicio de Informática de la Universidad Carlos III de Madrid. Estos objetivos se pueden resumir como: el desarrollo de una aplicación on-line de subtitulado que permita:

- **Reproducir los vídeos** existentes en el Gestor ARCAMM: debe incorporar un reproductor que permita mostrar los vídeos del Gestor y controlar su reproducción.
- **Generar un archivo de subtítulos:** este archivo debe contener la información que representa a cada subtítulo. Además, debe ser visualizado durante la reproducción del vídeo en cuestión, sincronizándose con éste, y ser editado cuando el usuario lo desee.
- **Permitir la integración** en el Gestor ARCAMM: la aplicación debe presentar una interfaz compatible con la interfaz del Gestor y posibilitar una comunicación con él para poder intercambiarse información necesaria para el proceso de subtitulación.

Para ello, se debe cumplir los siguientes objetivos específicos:

- **Editor con interfaz amigable:** debe presentar los elementos de forma que se identifiquen claramente y facilitar la edición de subtítulos presentando iconos o botones que ayuden al subtitulador a realizar su trabajo.
- **Reproductor con interfaz amigable:** el interfaz del reproductor también debe poseer botones de fácil identificación y presentar aquellos elementos necesarios para un buen control de la reproducción de los vídeos.
- **Activación/desactivación de subtítulos:** el reproductor debe presentar una opción que habilite y deshabilite la presentación de los subtítulos a elección del usuario.
- **Gestión de formato y control de errores:** la aplicación debe proporcionar la posibilidad de modificar algún aspecto del formato del texto de los subtítulos y controlar errores que puedan producirse durante el proceso de subtitulación.
- **Esquema de representación de subtítulos:** la aplicación debe generar un archivo de subtítulos que siga un esquema definido a tal fin y compatible con el reproductor. Este esquema debe ser definido con anterioridad y debe representar a los datos que definen a cada subtítulo.
- **Importación de otros formatos de subtítulos:** para proporcionar compatibilidad con otras herramientas de subtitulación, la aplicación deberá permitir la importación de archivos de subtítulos. En el proceso, el archivo importado se convertirá al esquema del archivo de subtítulos soportado por la aplicación.
- **Acceso a través de ARCAMM:** el acceso a la aplicación estará integrado con el sistema de autenticación empleado por ARCAMM.

Para conseguir estos objetivos, se aplicará un ciclo de vida en cascada con las siguientes fases: obtención de los requisitos mediante reuniones con el cliente, en este caso, el equipo del Área de Audiovisuales del Servicio de Informática de la Universidad Carlos III de Madrid; análisis de los requisitos; evaluación de las tecnologías a utilizar y definición del modelo arquitectónico de la aplicación; desarrollo de la aplicación y pruebas de su funcionamiento; y despliegue y puesta en producción.

1.3. Estructura del proyecto

En este capítulo se ha mostrado una introducción al proyecto, resumiéndose las motivaciones de la puesta en marcha de la aplicación a desarrollar y describiéndose los objetivos que se deben cumplir.

En el *Capítulo 2. Estado del arte* se proporcionará una visión del estado actual de los principales aspectos relacionados con el desarrollo de este proyecto, principalmente, Accesibilidad Web para contenidos audiovisuales y Subtitulado.

En el *Capítulo 3. Herramientas para la elaboración del proyecto*, se describirán las herramientas empleadas para el desarrollo del proyecto, tanto tecnologías utilizadas, como lenguajes de programación y entornos de trabajo.

En el *Capítulo 4. Desarrollo del proyecto* se describirán las distintas fases y artefactos generados dentro del proceso de desarrollo de la aplicación.

En el *Capítulo 5. Conclusiones* se incluirán las conclusiones del proyecto y las recomendaciones para mejorar el diseño y funcionamiento de la aplicación en un futuro.

Por último, se incluye las referencias bibliográficas en el *Capítulo 6. Bibliografía*, y algunas definiciones en el *Capítulo 7. Glosario*.

CAPÍTULO 2. ESTADO DEL ARTE

El objetivo de este capítulo es describir los conceptos necesarios para comprender mejor en qué consiste el presente proyecto y conocer su contexto.

Para empezar, se definen los conceptos de contenido multimedia y contenido audiovisual y la relación existente entre ambos términos.

En el siguiente apartado, se explicará qué son los Federadores de Contenido Multimedia y los Gestores de Contenido Multimedia, además de describir brevemente los Gestores y Federadores más importantes.

Posteriormente, se presentará el concepto de *Accesibilidad Web* y la accesibilidad de los contenidos multimedia en una web, pasando, después, a hablar de la subtitulación y las características que presenta.

A continuación, se detallarán las herramientas de subtitulación más conocidas, comentando sus características y comparando entre ellas los aspectos más importantes que deben presentar estas herramientas.

Por último, se hablará del lenguaje unificado de modelado (*Unified Modeling Language*, UML), utilizado en la parte de desarrollo del proyecto para identificar los requisitos funcionales del sistema a desarrollar.

2.1. Gestión de Contenido Multimedia

2.1.1. El contenido multimedia y audiovisual

Multimedia [Lamarca, 2009] es la combinación o utilización de dos o más medios de forma concurrente. En principio, la cualidad multimedial no está restringida a la informática, sin embargo, poco a poco, el término multimedia se ha ido generalizando para referirse a la informática y al mundo digital. Los medios que suelen combinarse son: texto, imágenes estáticas, vídeo, animaciones y audio.

Actualmente, el término multimedia se ha ampliado uniéndolo al de interactividad. Esto se produce cuando el usuario tiene cierto control sobre la presentación del contenido, como qué desea ver y cuándo desea verlo. De aquí nace el concepto de *Hipermedia*, aplicado sobre todo a un soporte abierto u on-line, que es la unión de hipertexto y multimedia.

Según Lamarca, [Lamarca, 2009] la hipermedia conjuga tanto la tecnología hipertextual, como la multimedia. Si la multimedia proporciona una gran riqueza en los tipos de datos, el hipertexto aporta una estructura que permite que los datos puedan presentarse y explorarse siguiendo distintas secuencias, de acuerdo a las necesidades y preferencias del usuario.

Así, cuando un programa, un documento o una presentación combinan adecuadamente los medios, se mejora notablemente la atención, la comprensión y el aprendizaje, ya que se acerca a la manera habitual en que los seres humanos nos comunicamos.

El contenido multimedia encuentra su uso en muchas áreas que no tienen porqué estar relacionadas del todo con las nuevas tecnologías, como el arte, educación, entretenimiento, ingeniería, medicina, matemáticas, negocio o la investigación científica.

Con la definición de multimedia, y para el desarrollo de este proyecto, se hace necesario aclarar otro concepto: el contenido *audiovisual*. Muchas veces, estos términos se confunden y, a veces, se usan indistintamente, pues su significado no está del todo claro.

Según la Real Academia Española (RAE), audiovisual se define como “Que se refiere conjuntamente al oído y a la vista, o los emplea a la vez. Se dice especialmente de métodos didácticos que se valen de grabaciones acústicas acompañadas de imágenes ópticas”, esto es, se refiere a todo lo relativo al uso simultáneo de lo auditivo y de lo visual. Sin embargo, audiovisual también se emplea de una forma más amplia (a la que estamos más acostumbrados), en la que el término incluye el uso, no sólo simultáneo, sino también alternativo de lo auditivo y de lo visual, con lo que, por ejemplo, una película muda también entraría en la categoría de audiovisual.

Pero, ¿cuál es la relación o diferencia entre ambos conceptos? Para verlo, se puede hacer una clasificación de los contenidos multimedia, incluida en la *Figura 2.1*:

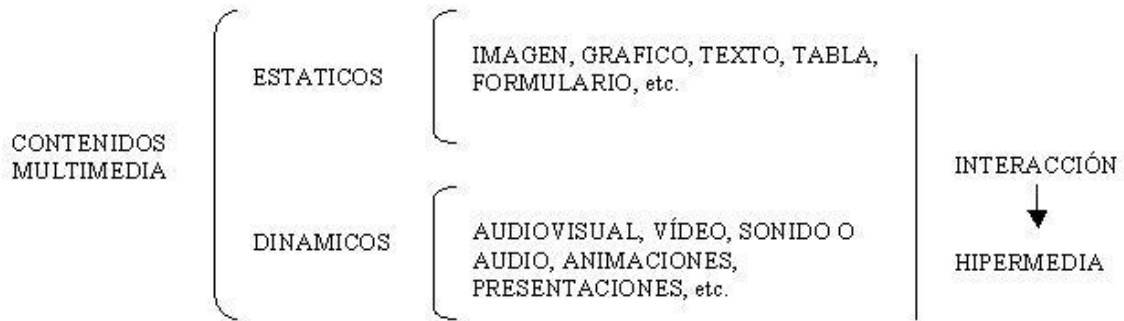


Figura 2.1. Clasificación de contenidos multimedia

Los contenidos multimedia dinámicos son aquellos a los que se les puede asociar una duración temporal, como el vídeo, el audio, etc. Por el contrario, los contenidos estáticos no poseen información que describa su duración, entre ellos tenemos las imágenes estáticas, el texto, etc. Si todos estos contenidos se asocian a la interacción, tenemos la hipermedia, como se explicó anteriormente.

Según esta clasificación, que suele emplearse para diferenciar los conceptos, audiovisual se incluye en el multimedia dinámico, aunque, como se ha dicho antes, el vídeo y sonido por separado se podría incluir como audiovisual. Así, para tener una visión más general, en este documento, se hará referencia al contenido multimedia dinámico como contenido audiovisual.

2.1.2. Gestores de Contenido Multimedia

Un Gestor de contenido multimedia es una aplicación que permite la catalogación, almacenamiento, clasificación, gestión y, en algunos casos, la distribución del contenido multimedia que genera una institución en concreto:

- **Catalogación:** el usuario sube al servidor los contenidos multimedia y añade información que describe a dicho contenido.
- **Almacenamiento:** los contenidos multimedia se guardan en una base de datos, además de la información que los describe. Este conjunto de información se denomina *metadatos*: información estructurada que describe los contenidos multimedia, como el nombre, el tamaño, tipo de dato, dónde se encuentra, categoría a la que pertenece, etc.
- **Clasificación:** los contenidos multimedia se clasifican en distintas categorías para una búsqueda más óptima. Cada institución elige las categorías dependiendo de sus contenidos o del uso que le vayan a dar a su Gestor.
- **Gestión:** mediante un perfil de administrador, el Gestor permite editar los contenidos multimedia, borrarlos, añadir o modificar los metadatos, etc.
- **Distribución:** algunas instituciones necesitan enviar su contenido multimedia para distribuirlo por Internet, aunque lo más común es que sean los Federadores los que se encarguen de ello. Para distribuir contenidos multimedia existen varias técnicas como: *streaming*, vídeo bajo demanda, *podcast* (ver *Capítulo 7. Glosario*), etc.

Además, los Gestores muestran los contenidos multimedia y los metadatos a través de una interfaz.

Se puede decir que ciertos contenidos de los Gestores pueden “alimentar” a los Federadores ya que éstos muestran el contenido multimedia que las instituciones desean publicar de aquellos contenidos que almacenan en sus Gestores.

A continuación, se describen los Gestores más importantes en la actualidad.

2.1.2.1. PuMuKIT

PuMuKIT (PUBlicador MULTimedia en KIT) [pumukit, 2009] es una aplicación desarrollada por la Universidad de Vigo que permite publicar, vía Internet, los contenidos audiovisuales producidos en una institución. PuMuKIT publica los contenidos multimedia almacenados en su base de datos de dos modos diferentes:

- Generando un portal web, una **WebTV** (ver *Capítulo 7. Glosario*): cuenta con un área para transmisiones en directo y una Mediateca (ver *Capítulo 7. Glosario*) en la que se puede seleccionar el canal temático de interés y acceder al vídeo de la conferencia, clase o acto que se desee ver.
- Creando **flujos RSS** (ver *Capítulo 7. Glosario*) compatibles con ARCA: este flujo de datos (en formato RSS compatible con el RSS que soporta ARCA) permite que la base de datos, con todos los contenidos audiovisuales de una institución, se sume a una gran base de datos central, el servidor ARCA.

ARCA es un Federador que surge del Área de Audiovisuales del Servicio de Informática de la Universidad Carlos III de Madrid, del cual se hablará en el apartado 2.1.3.4. ARCA.



Figura 2.2. Ejemplo de PuMuKIT en la Universidad de Vigo (tv.uvigo.es)

2.1.2.2. MediaMosa

MediaMosa [MediaMosa, 2009] es una plataforma de gestión y distribución de contenido multimedia. Esta plataforma es de código libre, escalable, de alta calidad y está basada en el Gestor de contenidos Drupal [Drupal, 2010].

Permite acceder a repositorios de datos, de metadatos y a servidores de codificación y streaming y ofrece un acceso a la aplicación de dos modos: a través de una interfaz de usuario (sólo para visualizar los contenidos) o de administrador (para gestionar la información). Además, permite gestionar cualquier tipo de contenido multimedia como: audio, vídeo, documentos, etc.

Leraar24 [Inloggen mijn Video's](#)

Blijvend in ontwikkeling

Home Onderwerpen Forum Links Doe mee!

Zoeken: Alle onderwijssectoren ☒ Gebruikers video's

Deze week onder de aandacht bij Leraar24

Studio24... Ronald Plasterk
 nieuw Wat heeft Ronald Plasterk als minister van onderwijs bereikt?
 17:18 min. ★★★★★ MBO PO VO 0 222

De Kwestie - School-oudercontract
 Moet je afspraken tussen ouders en school contractueel vastleggen?
 07:02 min. ★★★★★ MBO PO VO 0 1130

Breed Onderwijsdebat introductie
 nieuw Wat is de visie van de politieke partijen over de toekomst van het onderwijs? Een debat tussen politici en leerlingen.
 09:40 min. ★★★★★ MBO PO VO 0 195

Wat is Leraar24
 Leraar24 is een online platform van, voor en door leraren. Met praktische video's en inhoudelijke dossiers worden vraagstukken uit het onderwijs behandeld.
[Bekijk hier de promo >>](#)

Volg ons op twitter

Nieuwe dossiers

De man voor de klas in het po
 Nog maar 15% van de basisschoolleraars is man. Hoe komt dit en wat is er...

De Jos van Kemenade Award
 De Jos van Kemenade Award heeft als doel scholen te stimuleren om een go...

[Bekijk alle nieuwe dossiers](#)

Nieuwe video's

Professionalisering van leraren
 Wat kunnen u en uw leidinggevenden doen om goed te professionaliseren.

Informaticaproject: It's Academy
 Hoger en voortgezet onderwijs werken samen bij de bètavakken om het lera...

[Bekijk alle nieuwe video's](#)

Actueel

Tips of advies nodig?
 Word lid van de community Lerarengilde!
[Direct online advies!](#)

Upload uw video op Leraar24
 Heeft u een inspirerende of instruerende video die u graag wilt delen met uw collega's? Of heeft u ideeën voor nieuwe video's? Dan is Leraar24 dé plek om dit te delen!
[Doe mee >>](#)

Leraar24.nl is een samenwerking tussen Kennisnet, Ruud de Moor Centrum, SBL & Teleac | [Disclaimer](#) | [Gebruikersreglement](#) | [Over leraar24](#) | [Contact](#)

Figura 2.3. Ejemplo de MediaMosa

Las principales características de MediaMosa son:

- Plataforma de distribución de audio y vídeo (además de otros contenidos).
- *Streaming* de varios formatos, como Flash [Adobe, 2009], H.264/MPEG-4 [ITU, 2005] y WindowsMedia [Microsoft, 2010].
- Transcodificación basada en FFmpeg¹.
- Código sujeto a la licencia libre GNU Public License v2 [GNU, 2010].

2.1.2.4. DSpace

DSpace [Rodríguez, 2008] es un software de código abierto diseñado por el *Massachusetts Institute of Technology* (MIT) y los laboratorios de *Hewlett-Packard* (HP Labs).

¹ FFmpeg es un software de código libre para grabar, convertir y hacer streaming de audio y vídeo. FFmpeg está desarrollado en GNU/Linux, pero puede ser compilado en la mayoría de los sistemas operativos, incluyendo Windows.

Se encarga de gestionar repositorios de ficheros (textuales, audio, vídeo, etc.), facilitando su depósito, organizándolos en comunidades, asignándoles metadatos y permitiendo su difusión a federadores o agregadores.

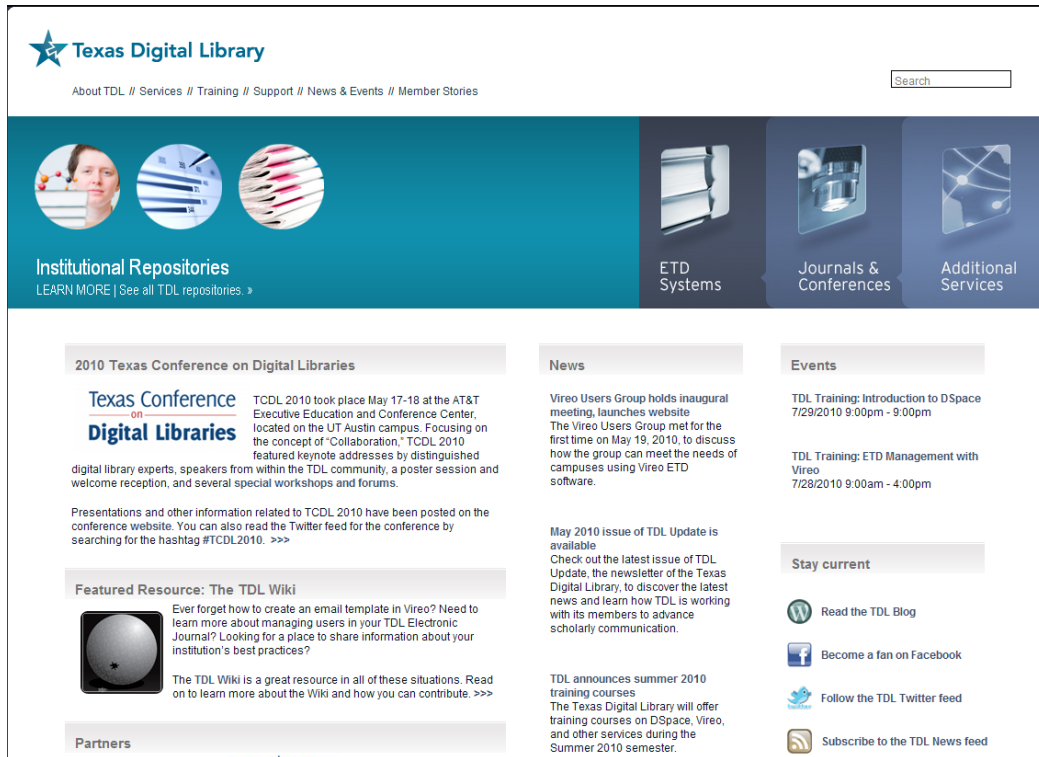


Figura 2.4. Ejemplo de DSpace en la librería digital de Texas (www.tdl.org)

Las características más importantes que presenta son [dspace, 2008]:

- Personalizable para las necesidades requeridas en cada caso.
- Almacena y preserva gran variedad de contenido digital.
- Usado por instituciones educativas, gubernamentales, privadas y comerciales.
- Fácil de instalar y de usar.

Estas características han hecho que sea un programa muy utilizado por las instituciones académicas para gestionar el repositorio donde los investigadores depositan sus publicaciones y materiales de búsqueda, con objeto de darles una mayor visibilidad.

2.1.2.5. ARCAMM: el gestor de la Universidad Carlos III de Madrid

ARCAMM (ARCA *Media Manager*) es un portal **IPTV** (ver Capítulo 7. Glosario) desarrollado por el Área de Audiovisuales del Servicio de Informática y Comunicaciones de la Universidad Carlos III de Madrid. ARCAMM [Cruz, 2009] es un complemento al portal de ARCA que permite la publicación de los contenidos multimedia que soporta el federador.

ARCAMM es un sistema basado en el acceso vía web [UC3M, 2009 a] que permite acceder a todos los contenidos multimedia (audio, vídeo, documentos,...) generados por la Universidad Carlos III de Madrid a través del Área de Audiovisuales.

El portal permite almacenar y catalogar, basado en distintas categorías, los contenidos multimedia. Se pueden catalogar vídeos en los formatos Windows Media y Flash para ser distribuidos por la red mediante *streaming*. También, vídeos en formato .mp4 para ser descargados mediante *podcast* para, por ejemplo, poder ser visualizados en dispositivos móviles (ipod, pda, etc.).

Además de los archivos de audio y vídeo, se puede adjuntar otro tipo de ficheros como: .pdf, .doc, .ppt, etc. El portal también permite visualizar emisiones en directo de aquellos actos en los que se requiera esta modalidad.

ARCAMM admite la publicación de los contenidos de dos maneras: pública y privada. Los contenidos públicos pueden ser vistos por cualquier persona que utilice el Gestor y los privados, sólo por aquellas que posean un usuario y contraseña de administrador del portal. Además, todo aquello que se marca como público es de manera inmediata publicado en el portal de ARCA.

Otra característica de ARCAMM es que da la posibilidad de suscribirse a un canal o a una categoría. Suscribirse permite estar informado de nuevos anuncios de contenido que aparezcan en el portal del federador ARCA. Así, se puede recibir automáticamente las novedades de un canal, de una categoría o de palabras clave de interés para el suscrito, sin necesidad de entrar en el portal o disponer de navegador,

ya que con un programa compatible con RSS (como iTunes) o con un lector de RSS, se puede acceder a la información.

Universidad Carlos III de Madrid

audiovisuales

Hoy es: **Martes, 08 Junio 2010.**
Mira los actos programados para este mes.

arcamm
Media Manager

Inicio Eventos Videos/Podcast Series/Cursos RSS ARCA-RSS

Contenidos Multimedia
2469.Horas 5739.

Se está migrando el contenido audiovisual desde el antiguo repositorio al nuevo. Si no encuentras el video que buscas, puedes probar a buscarlo [aquí](#)

UC3M TV Solemn Acto de Apertura del Curso Académico 2009/2010

Programación para hoy
No existen eventos programados para hoy.

Programación semanal

X 09	J 10	V 11	S 12	D 13	L 14	M 15

Los videos más nuevos
Premios de Excelencia 2010 Acto de Entrega

Canal: Universidad Carlos III de Madrid
Descripción: Entrevista al Rector de la UC3M, Daniel Peña, y al Presidente del Consejo Social y Vicepresidente Tercero de Banco Santander, Matías Rodríguez Inciarte, con motivo del acto de entrega de los Premios de Excelencia 2010.
Fecha del contenido: 2010/05/26
Num. de veces visto desde el portal: 22

PREMIOS DE EXCELENCIA '10
CONSEJO SOCIAL DE LA UNIVERSIDAD CARLOS III DE MADRID

Programación Mensual

		Anterior Junio 2010 Siguiente						
L	M	X	J	V	S	D		
	1	2	3	4	5	6		
7	8	9	10	11	12	13		
14	15	16	17	18	19	20		
21	22	23	24	25	26	27		
28 (1)	29 (1)	30						

Más información acerca de las categorías

- > **Ciencias de la Salud (34)**
Farmacia, Medicina, Enfermería, Fisioterapia, Logopedia, Odontología...
- > **Ciencias Experimentales (19)**
Estadística, Matemáticas, Biología, Física, Química, Geología...
- > **Ciencias Sociales y Jurídicas (463)**
Derecho, Economía, Pedagogía, Periodismo, Psicología, Turismo...
- > **Enseñanzas Técnicas (549)**
Arquitectura, Aeronáutica, Telecomunicaciones, Informática, Industriales...
- > **Humanidades (1342)**
Bellas Artes, Filología Hispánica, Filosofía, Geografía, Historia...

Figura 2.5. Portal de ARCA-MM en la Universidad Carlos III de Madrid
(marge2.uc3m.es/arcamm)

2.1.3. Federadores de Contenido Multimedia

Además de los Gestores, existe otra forma de acceder a contenidos multimedia: los Federadores. Un Federador de contenido multimedia [Spink, 2006] es una aplicación para la catalogación, clasificación, y distribución del contenido multimedia

que generan diversas instituciones (docentes, comerciales, etc.). Al igual que los Gestores, muestran los contenidos multimedia a través de una interfaz.

La característica principal de la federación es que la gestión de los contenidos se realiza de manera descentralizada, es decir, la institución propietaria del contenido es la encargada de gestionar su propia información y el federador sólo se encarga de las funciones mencionadas anteriormente.

Por el contrario, como ya se comentó, los Gestores sí se ocupan de la gestión de la información que tratan, ya que ésta es propiedad de la institución que utiliza el Gestor.

Como se explicó en el apartado 2.1.2. *Gestores de Contenido Multimedia*, los Federadores se alimentan del contenido de los Gestores. Para ello, el Gestor se comunica con el Federador a través de los metadatos. Una vez recibidos los metadatos, el Federador publica en su portal los contenidos referidos a esos metadatos.

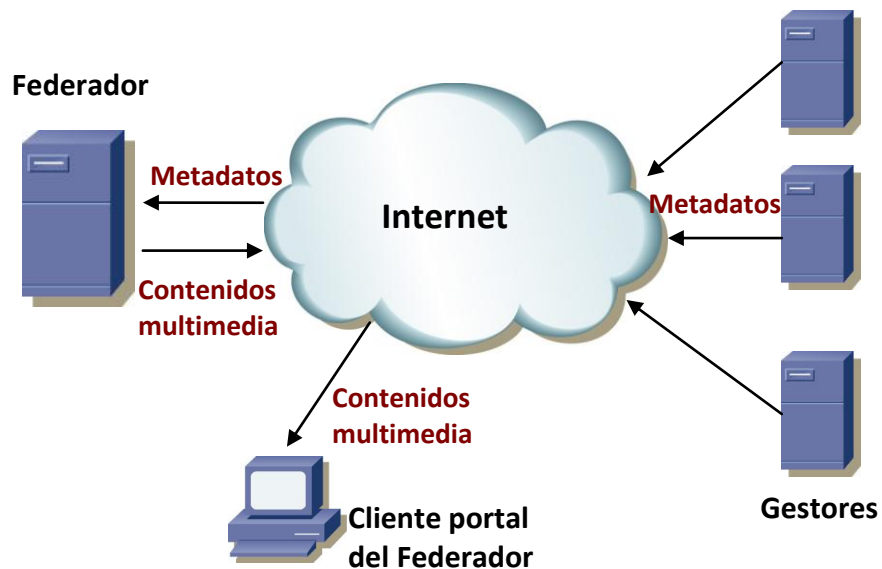


Figura 2.6. Comunicación entre Federador y Gestores

Para poder publicar contenidos en un Federador, el documento con los metadatos que envían los Gestores tiene que cumplir dos condiciones:

- Ser compatible con el documento RSS que soporta el Federador.

- Pertenecer a una de las categorías en las que el Federador clasifica los documentos RSS. Estas categorías son diferentes en cada Federador.

A continuación, se detallan los Federadores más importantes en la actualidad.

2.1.3.1. Steeple

Steeple [Steeple, 2010] es un comité para la creación de un software de federación de información fundado por la comunidad universitaria del Reino Unido, dirigida por la Universidad de Oxford, la Universidad de Open y la Universidad de Cambridge.

El objetivo del proyecto Steeple es investigar, desarrollar y documentar el sostenimiento de la infraestructura institucional para promover el *podcast* de los contenidos educativos de las universidades. Steeple también ha iniciado una sociedad para proporcionar soluciones escalables y de nivel empresarial, adecuadas al sector universitario en el Reino Unido, en las áreas de captura, proceso y envío de vídeo y audio.

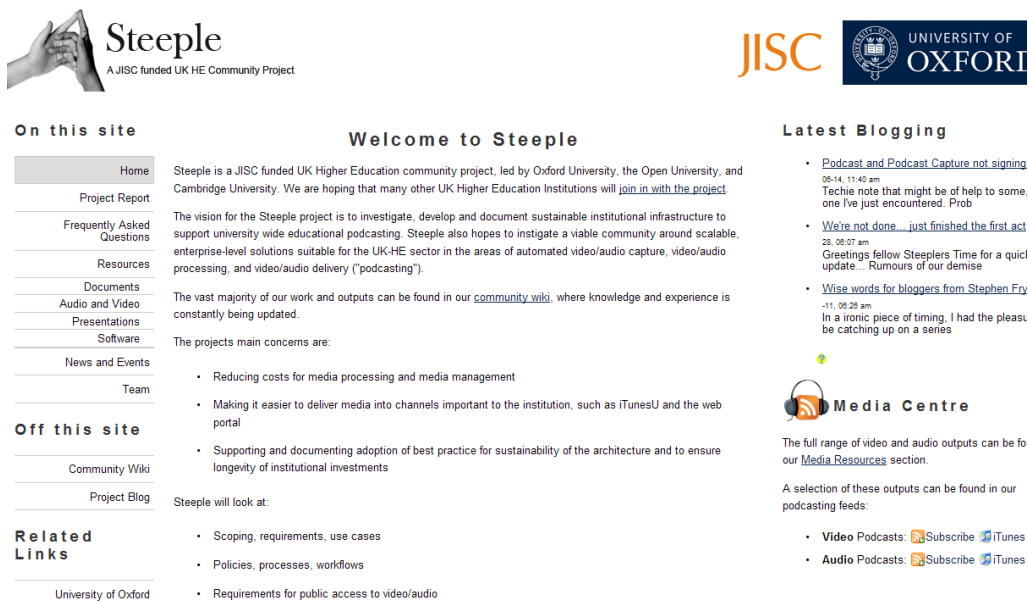


Figura 2.7. Captura de pantalla de Steeple

Los principales cometidos de este proyecto son:

- Reducir los costes para el proceso y gestión de multimedia.
- Facilitar la distribución de contenido multimedia a otras plataformas que sean importantes para cada institución.
- Apoyar y documentar la creación de una plataforma cuya arquitectura asegure la perdurabilidad en el tiempo de los contenidos de las instituciones.

2.1.3.2. iTunes U

iTunes U [Apple emagister, 2009] es una iniciativa de Apple, nacida en el marco de iTunes, para ayudar a derribar las barreras que limitan el aprendizaje. iTunes U es un medio a través del cual universidades e instituciones educativas en general pueden ofrecer contenidos de audio y vídeo a alumnos, ex-alumnos y, en general, a cualquier persona.

iTunes U utiliza la tecnología de la iTunes Store para que los usuarios puedan buscar, descargar y reproducir material educativo en formato digital.



Figura 2.8. Captura de pantalla de iTunes U

Las características principales que presenta son[Apple, 2010]:

- Está dirigida a estudiantes e instituciones.
- Interfaz fácil de usar e intuitiva.
- Accesibilidad para todos los estudiantes, inclusive a aquellos con problemas visuales, ya que emplea lectores de pantalla (ver *Capítulo 7: Glosario*).
- Permite elegir entre un acceso privado, sólo para los miembros de la comunidad educativa que haya creado el sitio, o público, donde los contenidos están accesibles a todo el mundo a través de la iTunes Store.

2.1.3.3. *Agrega*

Agrega [agrega, 2009] es un federador de repositorios de contenidos digitales educativos. Estos contenidos están validados pedagógicamente. Agrega es de acceso gratuito y está pensado para la comunidad docente.

El objetivo de Agrega es facilitar a la comunidad educativa una herramienta útil para integrar las Tecnologías de la Información y la Comunicación en el aula. Con ello pretende conseguir un acceso a cualquier contenido, desde cualquier lugar, utilizando distintos criterios de búsqueda.

Las características del proyecto son las siguientes:

- Impulsar un nuevo modelo de generación y aplicación de contenidos digitales al proceso de enseñanza y aprendizaje en el aula, e impulsar el desarrollo de una industria de producción de contenidos digitales en línea.
- Aunar los esfuerzos de generación y aplicación de contenidos digitales en línea y sentar las bases para potenciar la generalización del empleo de dichos contenidos en las aulas del sistema educativo español.
- El proyecto está dirigido a miembros de la comunidad educativa, con especial énfasis en los profesores y alumnos de enseñanza reglada no universitaria sin conocimientos especializados de tecnología.



Figura 2.9. Captura de pantalla de agrega

2.1.3.4. ARCA

ARCA [UC3M, 2009 b] es un proyecto, que surge del Área de Audiovisuales del Servicio de Informática de la Universidad Carlos III de Madrid, creado para federar la información relativa a los contenidos multimedia y emisiones vía streaming que ofrecen los integrantes de la red académica y de investigación nacional RedIRIS². El objetivo principal es paliar la falta de medios actual para dar a conocer a la comunidad académica todos estos contenidos docentes que, actualmente, apenas tienen visibilidad en la Web.

² RedIRIS (www.rediris.es) es la red académica y de investigación española. Proporciona servicios avanzados de comunicaciones a la comunidad científica y universitaria nacional. Está financiada por el Ministerio de Ciencia e Innovación y es gestionada por la entidad pública empresarial Red.es, del Ministerio de Industria, Turismo y Comercio.



Figura 2.10. Logo de ARCA

ARCA es un agregador de *feeds* o documentos RSS: las instituciones elaboran documentos XML, con información (metadatos) sobre sus contenidos, y ARCA se encarga de recoger esta información automáticamente e integrarla en una base de datos para mostrarla a los usuarios de forma inteligible, navegable y susceptible de búsquedas. Para que ARCA pueda “entender” los documentos XML, éstos deben utilizar la especificación RSS 2.0.

proyecto arca
federar. unir por alianza,
liga, unión o pacto entre varios.

Hoy es: **Sábado, 19 Junio 2010.**
Mira los actos programados para este [mes](#).

Federación de metadatos sobre contenidos multimedia y retransmisiones programadas.

Red IRIS

Inicio Eventos Vídeos.Podcast Series.Cursos Miembros Documentación RSS

Contenidos Multimedia **11687**. Horas **7334**.

Buscar

I Jornadas Software Libre en la Enseñanza
UNIVERSIDADE DE VIGO

Programación para hoy
No existen eventos programados para hoy.

Programación semanal

D 20	L 21	M 22	X 23	J 24	V 25	S 26

Los videos más nuevos
[Mario Vargas Llosa Doctor Honoris Causa](#)

Miembro: Universidad de Castilla-La Mancha
Descripción: Presentación
Localización: Paraninfo del edificio Melchor de Macanaz, Campus de Albacete
Categoría: Humanidades
Fecha del contenido: 2010-06-18
[Mario Vargas Llosa Doctor Honoris Causa](#)

Miembro: Universidad de Castilla-La Mancha
Descripción: Intervención de D. José María Barreda
Localización: Paraninfo del edificio Melchor de Macanaz, Campus de Albacete

Programación Mensual

Anterior		Junio 2010							Siguiete	
L	M	X	J	V	S	D				
	1	2	3	4 (1)	5 (1)	6				
7	8	9	10	11	12	13				
14	15	16	17	18	19	20				
21	22	23	24	25	26	27				
28 (1)	29 (1)	30								

Miembros: Todos
Domínios: Todos
Categorías: Todas

Figura 2.11. Captura de pantalla de ARCA
(arca.rediris.es)

2.2. Accesibilidad Audiovisual

En los últimos años, se ha presenciado una gran evolución de Internet y las tecnologías. Ahora, la Web nos ofrece aplicaciones de todo tipo y, gracias a ello, ha crecido el uso de contenido audiovisual, el cual está presente en la mayoría de sitios web. Sin embargo, la mayoría de estos contenidos no son accesibles para todos los usuarios.

Cuando se habla de accesibilidad, [AENOR, 2009] suele asociarse a personas con discapacidad pero este concepto incluye a cualquier persona ya que también se consideran otros factores como los recursos tecnológicos, sobre todo, en los contenidos audiovisuales, donde éstos toman gran relevancia.

Por tanto, es importante proporcionar un acceso universal a los contenidos en una web y realizar un diseño de accesibilidad que sea independiente de las características de acceso de cada usuario y de los contextos de uso.

2.2.1. Accesibilidad Web

2.2.1.1. Definición

Para definir la Accesibilidad Web, hay que definir primero la accesibilidad como concepto general. Así, se tiene que la accesibilidad [AENOR, 2009] es el grado en el que algo puede ser usado, visitado o accedido por todas las personas, independientemente de sus capacidades técnicas, cognitivas o físicas.

Basándose en esta definición y extendiéndola al uso de Internet, surge la *Accesibilidad Web*, la cual define el consorcio internacional para la *World Wide Web*, el *World Wide Web Consortium* (W3C) [W3C, 2008 a]: “Hablar de Accesibilidad Web es hablar del acceso de todos a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, localización geográfica y capacidades de los usuarios”.

Con esto se tiene que la Accesibilidad Web [W3C, 2005] facilita el acceso a contenidos de la Web y beneficia, tanto a personas con cualquier tipo de discapacidad (visual, cognitiva, auditiva...), como a personas sin discapacidad que, por ejemplo, presenten problemas al acceder a la Web (por ejemplo, por conexiones lentas), alguna incapacidad transitoria y/o personas de edad avanzada.

2.2.1.2. Estandarización

El W3C se encargó de estudiar la Accesibilidad Web desarrollando La Iniciativa de Accesibilidad Web (*Web Accessibility Initiative: WAI*), actividad cuyo objetivo [W3C, 2005] es desarrollar pautas y técnicas que proporcionen soluciones accesibles para el software web y para los desarrolladores web. También lleva a cabo una labor de educación y concienciación de la importancia que tiene el diseño accesible de páginas web.

Las principales áreas de trabajo que estudia WAI, [Moreno, 2008] junto con otras organizaciones de todo el mundo, para conseguir la Accesibilidad Web son: tecnología, directrices, herramientas, formación y difusión, e investigación y desarrollo.

De los trabajos desarrollados por la WAI destacan las pautas de Accesibilidad Web para diferentes componentes [W3C, 2006 c]:

- ***Pautas de Accesibilidad al Contenido en la Web (WCAG)***: donde se tratan temas de contenido web. Son utilizadas por desarrolladores, herramientas de autor y herramientas de evaluación de accesibilidad.
- ***Pautas de Accesibilidad para Herramientas de Autor (ATAG)***: que tratan las herramientas de autor. Estas herramientas son software que se emplea para crear páginas y contenido web.
- ***Pautas de Accesibilidad para Herramientas de Usuario (UAAG)***: donde se habla de los navegadores web y reproductores multimedia, incluyendo algunos aspectos de tecnologías asistivas (ver *Capítulo 7: Glosario*).

Estas pautas [Moreno, 2008] son referencia en la mayoría de legislaciones sobre Tecnologías de la Información (TIC) pues son consideradas como normas oficiales de Accesibilidad Web en la Unión Europea. De hecho, en España, según la legislación vigente en el BOE de 2007³, los sitios web de las Administraciones Públicas y otros organismos deben ser accesibles según la Norma de la Asociación Española de Normalización y Certificación (AENOR) UNE 139803:2004 [INTECO, 2010], en la que hay una correspondencia con las pautas al contenido web WCAG.

2.2.2. Accesibilidad a los contenidos audiovisuales

2.2.2.1. Introducción

Debido al crecimiento que están experimentando los contenidos audiovisuales, este apartado se centrará en cómo ofrecer accesibilidad a dichos contenidos. Siguiendo las recomendaciones que indican las Pautas de Accesibilidad al Contenido en la Web (WCAG), para que pueda darse la accesibilidad a los contenidos audiovisuales, hay que fijarse en tres aspectos [Voces, 2008]:

- **Accesibilidad física:** engloba todos los factores relacionados con el acceso físico al dispositivo que se utiliza para acceder a la Web como, por ejemplo, la accesibilidad del teclado.
- **Accesibilidad lógica:** engloba todos los factores relacionados con el acceso al sistema operativo y a las aplicaciones que en él se ejecutan como, por ejemplo, la accesibilidad del navegador web o del reproductor multimedia.
- **Accesibilidad del contenido:** incluye todos los elementos de diseño utilizados en la/s página/s de un sitio web como, por ejemplo, los sistemas de navegación.

³ Boletín Oficial del Estado, Real Decreto 1494/2007, de 12 de noviembre, por el que se aprueba el Reglamento sobre las Condiciones Básicas para el Acceso de las Personas con Discapacidad a las Tecnologías, Productos y Servicios Relacionados con la Sociedad de la Información y Medios de Comunicación Social.

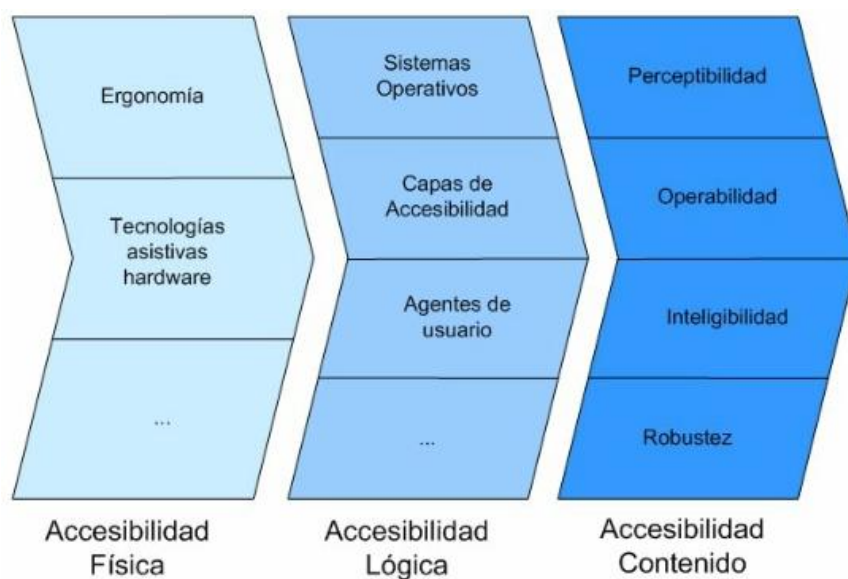


Figura 2.12. Modelo de requerimientos para la accesibilidad audiovisual

En relación con este proyecto, el aspecto más relevante es la accesibilidad del contenido, para el cual se utilizan, principalmente, las técnicas de subtítulo, audiodescripción y lengua de signos. De este aspecto se hablará en el apartado 2.2.2.3. *Principios de las WCAG 2.0 aplicados al contenido audiovisual.*

2.2.2.2. Las Pautas de Accesibilidad al Contenido en la Web (WCAG)

Como se comentaba anteriormente, en las WCAG se definen las recomendaciones a seguir para obtener contenidos accesibles en la Web. Existen dos versiones de estas pautas, la 1.0 y la 2.0. En la *Tabla 2.1* se muestran las diferencias entre las dos versiones [Moreno, 2008].

WCAG	Estructura	Puntos de verificación criterios de éxito	Niveles de adecuación
1.0 (1999)	14 pautas con 65 puntos de verificación	Los puntos de verificación tienen asignados un nivel de prioridad 1,2 ó 3	Nivel A, AA, AAA
2.0 (2008)	4 principios básicos con 12 pautas que contienen un total de 61 criterios de éxito	Los criterios de éxito tienen asignados un nivel de adecuación A, AA o AAA	Nivel A, AA, AAA

Tabla 2.1. Características de las WCAG 1.0 y las WCAG 2.0

WCAG 1.0 consiste en 14 pautas o directrices que proporcionan soluciones de diseño, utilizando como ejemplo situaciones comunes en las que el diseño de una página puede producir problemas de acceso a la información. Las Pautas contienen, además, una serie de puntos de verificación que ayudan a detectar posibles errores. Cada punto de verificación está asignado a uno de los tres niveles de prioridad establecidos por las pautas:

- **Prioridad 1:** son aquellos puntos que un desarrollador web tiene que cumplir ya que, de otra manera, ciertos grupos de usuarios no podrían acceder a la información del sitio web.
- **Prioridad 2:** son aquellos puntos que un desarrollador web debería cumplir ya que, si no fuese así, sería muy difícil acceder a la información para ciertos grupos de usuarios.
- **Prioridad 3:** son aquellos puntos que un desarrollador web debería cumplir ya que, de otra forma, algunos usuarios experimentarían ciertas dificultades para acceder a la información.

En función a estos puntos de verificación se establecen los niveles de conformidad [W3C, 2008 b]:



Nivel de Conformidad "A": todos los puntos de verificación de prioridad 1 se satisfacen



Nivel de Conformidad "Doble A": todos los puntos de verificación de prioridad 1 y 2 se satisfacen



Nivel de Conformidad "Triple A": todos los puntos de verificación de prioridad 1, 2 y 3 se satisfacen.

Si una página web alcanza alguno de estos niveles, pueden colocar el logo correspondiente, mostrados anteriormente, que el WAI ofrece para declarar la conformidad con las Directrices.

Por su parte, las WCAG 2.0 se organizan en **4 principios fundamentales**, los cuales marcan que un contenido es accesible si [Voces, 2008]:

1. Es perceptible: todas las personas son capaces de acceder a su contenido.
2. Es operable: todas las personas pueden interactuar adecuadamente con él.
3. Es inteligible: todas las personas son capaces de entender lo que contiene.
4. Es robusto: todas las personas pueden acceder a él independientemente del entorno operativo (sistema operativo, navegador, etc.) que tengan.

Estos principios conforman 12 pautas o directrices que no son comprobables, sino que proporcionan los objetivos básicos para hacer el contenido accesible, y sirven para comprender los criterios de éxito e implementarlos. Estas pautas [Carreras, 2008] se dividen en 60 criterios de éxito o puntos de comprobación, que sí son comprobables, y que equivalen a los puntos de verificación de la versión 1.0. Los puntos de verificación definen los niveles de adecuación A, AA o AAA, correspondientes a las prioridades de la versión anterior.

Es importante destacar que la 2.0 es la versión de las WCAG que se emplea como referencia actualmente.

2.2.2.3. Principios de las WCAG 2.0 aplicados al contenido audiovisual

Como se ha visto, los contenidos en una web deben cumplir cuatro principios. Estos principios conllevan distintas implicaciones dependiendo del contenido pero, en relación con este proyecto, se explicarán las implicaciones para los contenidos audiovisuales.

A continuación, se describen los principios de las WCAG 2.0 aplicados al contenido audiovisual. Como se resumen en [Voces, 2008]: para que el contenido sea perceptible, se necesitan las técnicas de subtítulo, audiodescripción y lengua de signos. Para que sea operable, el contenido debe ser interactivo, es decir, que permita una comunicación con el usuario. Inteligible, cuando se reproducen todas las connotaciones del contenido audiovisual y robusto si se emplean formatos y códecs compatibles con diferentes plataformas.

El contenido audiovisual perceptible

Los problemas de perceptibilidad asociados a un contenido audiovisual aparecen en las personas con discapacidades visuales o auditivas. Las técnicas utilizadas, principalmente, en estos casos son la audiodescripción, para las discapacidades visuales, y la subtitulación y la interpretación en lenguaje de signos, para las auditivas.

La audiodescripción consiste en añadir un contenido auditivo suplementario que se encarga de narrar toda aquella información que se transmite de forma visual y que resulta importante para su interpretación auditiva (por ejemplo, la descripción gestual de un actor). Para ello, el contenido de la audiodescripción se intercala aprovechando los silencios del contenido audiovisual.

En lo que se refiere a la subtitulación, consiste en añadir toda la información textual necesaria para poder interpretar la información visual de forma adecuada. Por ello, una subtitulación accesible no sólo consiste en visualizar los diálogos del contenido, sino también la inclusión de todos los sonidos que envuelven la escena (risas, truenos, sonidos de sirenas, etc.).

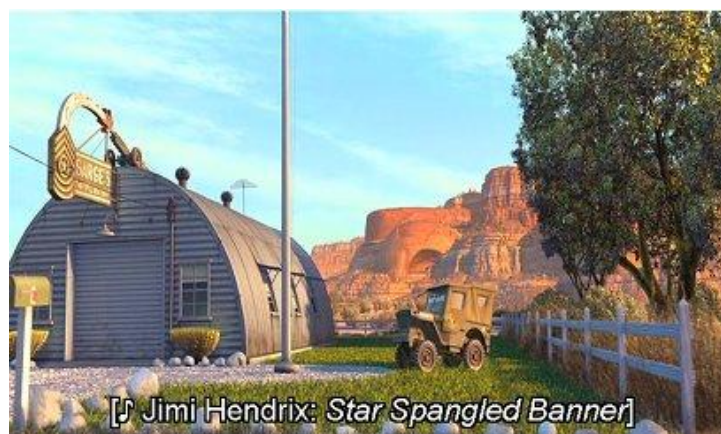


Figura 2.13. Ejemplo de subtítulo

Por último, se puede utilizar el lenguaje de signos. En general, consiste en añadir, en una pequeña área del contenido audiovisual, la representación del contenido en lengua de signos.

Actualmente suele ser una alternativa muy poco común en los entornos web. La razón fundamental es que, además de las dificultades tecnológicas, existen multitud de lenguas de signos y un conjunto relativamente pequeño de personas que utilizan este sistema de comunicación (entre un 6 y un 8 %).

La operabilidad del contenido audiovisual

Sólo tiene sentido hablar de la operabilidad del contenido cuando se trata de un contenido audiovisual interactivo. Actualmente existen tecnologías que permiten construir este tipo de contenidos y que son aprovechadas por los diseñadores para, por ejemplo, incorporar los controles de la reproducción.

Hay ocasiones en las que los diseñadores van más allá y aprovechan este tipo de contenidos para crear nuevas experiencias de usuario o para ofrecer nuevos productos como aventuras gráficas y otros juegos.

En estos casos, los requisitos de accesibilidad son:

- Todas sus funciones deben ser accesibles por teclado.
- Deben ser convenientemente descritas.
- Deben ser activables sus recursos de accesibilidad (subtitulado, audiodescripción o lengua de signos).

El contenido audiovisual inteligible

El contenido audiovisual es un contenido complejo, lleno de connotaciones y detalles no presentes en otros tipos de contenidos. Hacer inteligible un contenido audiovisual implica reproducir todas estas particularidades, con el máximo de neutralidad y objetividad.

Como se decía anteriormente, las técnicas disponibles para la accesibilidad audiovisual son la audiodescripción, la subtitulación y el lenguaje de signos. En todos los casos, se tiene una larga experiencia en el ámbito audiovisual tradicional (TV, cine, DVD, etc.) y

en España se disponen de normas técnicas UNE: la 153020:2005 [AENOR, 2005] para la audiodescripción, la 153010:2003 [AENOR, 2003] para la subtitulación y la 139804:2007 [AENOR, 2007] para el lenguaje de signos.

El contenido audiovisual robusto

No todos los contenidos audiovisuales están creados de la misma forma. En general, existen dos aspectos diferenciadores:

1. El **formato** o contenedor. Un contenido audiovisual puede estar formado por diversos canales de información: audio, video, subtítulos, información sobre estructura, etc. Un formato o contenedor es el encargado de definir el orden y la forma con la que se almacena toda esta información. Los archivos con extensión .avi, .mov o .mpg son ejemplos de formatos.
2. Los **códecs**. Seguramente un elemento común en todos los contenidos audiovisuales es su tamaño. Almacenar audio, y sobre todo vídeo, requiere de una gran cantidad de bytes que puede complicar su almacenamiento y su distribución. Los códecs son aplicaciones diseñadas para comprimir/descomprimir la información audiovisual (frecuentemente, a costa de la calidad de la misma) y facilitar su procesamiento. Por ejemplo, Divx es un códec de vídeo y MP3 un códec de audio.

De esta manera, sería posible tener un contenido multimedia en formato .avi, donde el video está codificado con Divx y el audio con MP3.

Desde el punto de vista de la accesibilidad audiovisual, es importante elegir formatos y códecs que estén presentes en la mayoría de plataformas para que cualquier persona pueda acceder al contenido audiovisual y evitar así problemas de compatibilidad como el que se representa en la *Figura 2.14*.

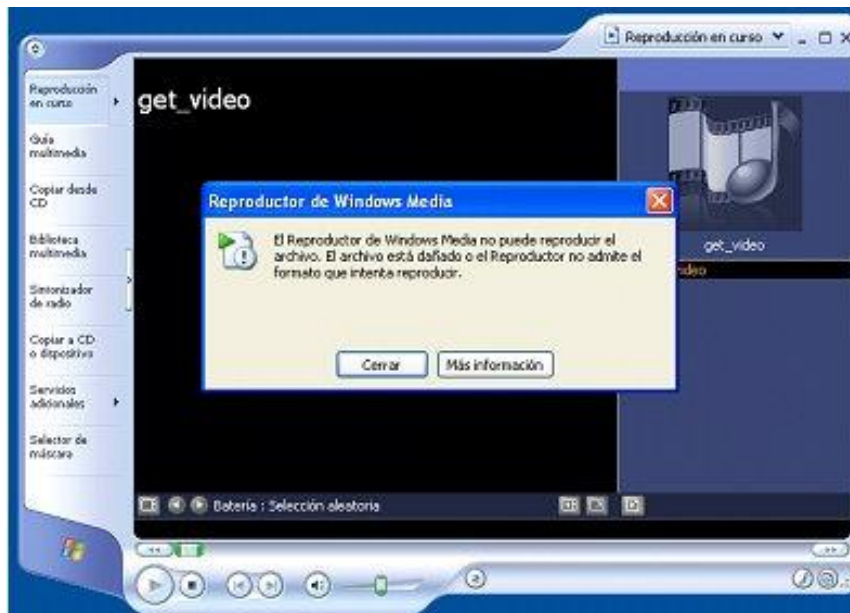


Figura 2.14. Error de códec

2.3. La subtitulación

La subtitulación es una técnica útil para facilitar la accesibilidad a los contenidos audiovisuales. Gracias a los subtítulos, las personas sordas y con discapacidad auditiva pueden acceder a la misma información que las demás personas, favoreciéndose la igualdad de oportunidades.

Pero del subtitulado también pueden beneficiarse otros usuarios [AENOR, 2003], ya que la subtitulación ayuda a mejorar las habilidades lecto-escritoras⁴. Entre estos usuarios tenemos a personas con dificultades lecto-escritoras, público infantil, personas que aprenden idiomas, etc.

La subtitulación consiste, en esencia, en determinar un *tiempo de entrada*, o tiempo en el que aparece el subtítulo, un *tiempo de salida*, o tiempo en el que

⁴ Lectoescritura, según la RAE: “Capacidad de leer y escribir”.

desaparece, y el texto que representará el audio del contenido audiovisual. Este audio no tiene por qué ser sólo los diálogos, como se explicará más adelante. Pero dependiendo de la manera en la que se realice la subtitulación o a las personas a las que vaya dirigida, aparecen varios tipos de subtitulación, donde los subtítulos presentan diferentes características.

Aspectos importantes de subtulado

- Se diferencian dos tipos de subtítulos: los subtítulos cerrados o “*close captions*” y subtítulos abiertos u “*open captions*”. La diferencia entre estos dos tipos radica en que los subtítulos cerrados se muestran a conveniencia del espectador, es decir, se tiene la opción de activarlos y desactivarlos. Sin embargo, los subtítulos abiertos aparecen siempre, ya que están insertados en la imagen, aunque la/s persona/s que estén viendo el contenido audiovisual no los necesiten.

Aunque los dos tipos son correctos, los subtítulos cerrados son los convenientes a usar para ofrecer una mayor accesibilidad.

- En la Televisión Digital Terrestre (TDT) se tienen dos subtitulados: el subtulado a través del Teletexto y el subtulado *Digital Video Broadcasting* (DVB). La diferencia fundamental entre estos subtítulos es que los subtítulos DVB, al ser en formato digital, poseen mayores prestaciones y una mejor calidad gráfica.



Figura 2.15. Subtítulos DVB (izquierda) y Subtítulos de Teletexto (derecha)

En la *Figura 2.15* se puede ver las diferencias entre los dos tipos de subtítulos. En este caso, los subtítulos DVB no presentan fondo negro (caja de subtítulos), aunque en otras ocasiones sí pueden tenerlos.

2.3.1. Tipos de subtitulación

2.3.1.1. Subtitulación literal o “Subtitles”⁵

Se basa en reproducir exactamente el discurso oral del vídeo o programa que se subtitula. Los subtítulos aparecen en la parte inferior de la imagen y la técnica consiste en transcribir los diálogos, como se ve, por ejemplo, en películas extranjeras en las se muestran los subtítulos de la lengua original para ayudar a comprenderla. Sin embargo, esta subtitulación no posee toda la información necesaria para personas sordas o con discapacidad auditiva grave.

2.3.1.2. Subtitulación adaptada para sordos o “Close Captions”

Esta subtitulación añade información adicional a la que aporta la subtitulación literal. Utiliza textos visuales para describir el diálogo, el ruido de fondo, los efectos de sonido e, incluso, emplea emoticonos (en inglés *smileys*) para mostrar el estado de ánimo de la persona que habla. Estos subtítulos se pueden ver, por ejemplo, en los que proporciona el Teletexto de las cadenas de televisión.

Las palabras habladas, las descripciones del ruido de fondo y efectos de sonido aparecen en diferentes lugares de la imagen: los diálogos en la parte inferior y los ruidos y efectos de sonido, en la esquina superior derecha. Además, emplea diferentes colores de letra (los principales, según la norma UNE 153010:2003, son: amarillo, verde, celeste y magenta), aparte del blanco que se emplea en la subtitulación literal, para diferenciar a las personas o personajes

⁵ En Estados Unidos y Canadá, se denomina *Close Captions* a la subtitulación adaptada para sordos y *Subtitles*, a la subtitulación literal.

protagonistas o cuyo número de diálogos es mayor. También, para identificar los efectos de sonido se utiliza un color de letra distinto al blanco.



Figura 2.16. Ejemplo de colores en subtítulos

Este subtítulo es más completo y el que se usa para proporcionar una mayor accesibilidad a los contenidos audiovisuales tanto en web, como en el resto de medios.

2.3.1.4. Subtitulación adaptada para personas con problemas de lecto-escritura

Esta técnica se basa en la modificación del texto del subtítulo para que pueda ser leído de manera comprensiva. Esta modificación se realiza cuando la locución posee una velocidad excesiva (el subtítulo debe permanecer un cierto tiempo para que, en general, pueda ser leído de manera comprensiva) o porque el subtítulo va dirigido a personas con problemas de lecto-escritura⁶.

Este tipo de subtitulación es más bien una manera de escribir el texto que conforma los subtítulos, ya que puede emplearse en las dos técnicas anteriores. Por ejemplo, si se adapta una película a una persona oyente con problemas de lecto-escritura, la subtitulación puede ser literal con los cambios necesarios para su

⁶ Entre las personas con problemas al leer y escribir se incluyen personas que poseen alguna dificultad a causa de una discapacidad, niños, personas que están aprendiendo un nuevo idioma, personas con un bajo nivel de estudios, etc.

comprensión o, si la adaptación es para una persona sorda con el mismo problema, se usará el subtitulado *close-caption* y los cambios en el texto pertinentes.

2.3.1.3. Subtitulación en directo

Se realiza al mismo tiempo que se emite el contenido audiovisual (eventos deportivos, programas de noticias, etc.) o se realizan congresos, conferencias, actos, etc. Se suele utilizar la *máquina de estenotipia* [AE Estenotipistas, 2010], máquina con la que se pueden ejecutar pulsaciones con una o más teclas que se presionan simultáneamente. La escritura posee las letras alfabéticas en caracteres de imprenta.

Los métodos o sistemas de estenotipia están diseñados sobre la base de un teclado de reducido número de teclas con los que se logra obtener con cada pulsación sílabas o palabras completas.



Figura 2.17. Máquina de estenotipia

En la actualidad, las máquinas de estenotipia que se emplean son informatizadas, es decir, son máquinas compatibles con la electrónica cuya característica principal es que realizan la transcripción en tiempo real ayudándose de un ordenador.

Otra forma de subtitular en directo es la subtitulación mediante reconocimiento de habla, para convertir la voz automáticamente en texto escrito. Esta técnica aún se está desarrollando pues el texto de los subtítulos no es del todo exacto, aunque en algunos sitios ya se ha puesto en marcha, como en el gestor de vídeos YouTube.

2.3.2. Lenguajes de subtitulación

A la hora de crear una subtitulación cerrada [Voces, 2008], se puede escoger entre dos lenguajes: el *Synchronized Multimedia Integration Language* (SMIL) [W3C, 2008 e], recomendación del W3C soportada por los reproductores multimedia más utilizados y el *Synchronized Accessible Media Interchange* (SAMI) [Microsoft, 2003], un formato propietario de Microsoft y soportado únicamente por *Windows Media Player*.

Cabe decir que el objetivo original de SMIL y SAMI no son específicamente la creación de subtítulos, sino más bien una forma de controlar la reproducción de los diferentes medios (audio, vídeos, textos, etc.) que componen una presentación.

2.3.2.1. SMIL

El lenguaje SMIL (pronunciado “smile”) fue desarrollado por el grupo *Synchronized Multimedia Activity* (SYMM) [W3C, 2010 c] del W3C para presentaciones multimedia interactivas donde el vídeo, audio, texto, gráficos y otros medios se combinan en tiempo real.

SMIL es un lenguaje basado en XML, fácil de aprender y comprender y puede escribirse usando un editor de texto sencillo. Los archivos SMIL se almacenan con la extensión .smil.

El SYMM ha desarrollado las versiones SMIL 1.0, SMIL 2.0, SMIL 2.1 y SMIL 3.0. En la actualidad sigue trabajando en la última versión, SMIL 3.0, que se convirtió en recomendación del W3C en diciembre del 2008. Esta versión tiene los siguientes objetivos [W3C, 2008 f]:

- Definir un lenguaje basado en XML que permita a los autores escribir presentaciones multimedia interactivas.
- Permitir la reutilización de la sintaxis y semántica de SMIL en otros lenguajes basados en XML, en particular aquellos que necesitan representar la temporización y sincronización. Por ejemplo, los componentes de SMIL se usan para integrar la temporización en XHTML [W3C, 2008 d] y SVG [W3C, 2010 b].

- Extender las funcionalidades contenidas en SMIL 2.1 en módulos nuevos o modificados.

2.3.2.2. SAMI

Este lenguaje tiene como objetivo simplificar el proceso de subtítulos, facilitando la accesibilidad universal del resultado. Dirigida a desarrolladores, educadores, productores y diseñadores multimedia, la especificación de SAMI está disponible al público como estándar abierto.

La tecnología SAMI amplía la capacidad de proporcionar subtítulos cerrados a una amplia gama de productos multimedia.

SAMI 1.0 fue diseñado y desarrollado para subtítulos el multimedia digital disponible en los sistemas de ordenadores. Los subtítulos SAMI coexisten con los contenidos multimedia como archivos separados de texto. Los subtítulos pueden ser fácilmente modificados, mantenidos y personalizados por diferentes idiomas.

SAMI es un lenguaje de etiquetado, fácil de aprender y editar, el cual puede escribirse, igual que SMIL, en un sencillo editor de texto. Las extensiones que utiliza son .smi y .sami.

2.3.3. Formatos de subtítulos

En la subtítulos cerrados, [Díaz, 2001] aparte de utilizar lenguajes para la creación de subtítulos, existen varios formatos que representan toda la información relativa a los subtítulos. Son archivos que almacenan esta información en formato de texto y que son leídos por el reproductor junto con el archivo de vídeo a subtítulos.

Algunos de los formatos son:

- *Timed Text (TT) Authoring Format 1.0 - DFXP*
- SubRip

- MicroDVD
- SubStation Alpha y Advanced SubStation Alpha
- Otros formatos: RealText, TXT.

2.3.3.1. Formatos más utilizados

SubRip

SubRip [Matroska, 2010 a] es un formato de archivo de subtítulos de texto plano compatible con la mayoría de reproductores de vídeo y de herramientas de edición de subtítulos. Su extensión es .srt y su contenido se compone de 4 partes:

- El número del subtítulo
- El tiempo en el que debe aparecer y desaparecer el subtítulo, con el formato *horas:minutos:segundos,milisegundos*.
- El texto del subtítulo
- Una línea en blanco que indica el comienzo de un nuevo subtítulo

También se pueden añadir etiquetas para dar estilo a la letra del texto, como cursiva o negrita (si se añaden etiquetas, deja de ser un formato de subtítulos plano).

MicroDVD

MicroDVD es un formato de archivo de subtítulos, para vídeo digital, compatible con un gran número de reproductores de vídeo. Un archivo MicroDVD posee la extensión .sub.

Su contenido consta de varias líneas que definen el texto de los subtítulos y los fotogramas del vídeo entre los que debe aparecer dicho texto. La duración de los subtítulos depende de los fotogramas por segundo del vídeo.

La sintaxis de cada línea es: *{fotograma de comienzo} {fotograma de fin} Texto*

Además, se pueden añadir etiquetas para dar formato al texto, como el estilo o el color de letra.

SubStation Alpha (SSA) y Advanced SubStation Alpha (ASS)

SubStation Alpha [Matroska, 2010 b] es un formato de archivo de subtítulos, basado en texto, que puede ser editado con cualquier editor de texto sencillo. Es más avanzado que los subtítulos planos como el SubRip ya que permite añadir estilos a los subtítulos. Su extensión es .ssa.

Advanced SubStation Alpha es similar al SSA pero con estilos más complejos. Su extensión es .ass.

El contenido de estos archivos consta de 3 partes:

- Información del archivo, como el título, quién lo ha creado, etc.
- Lista de estilos, en los que se puede definir la fuente, el tamaño, color del texto, estilo de fuente, espaciado entre caracteres, etc.
- Diálogos, en los que se indica el tiempo de entrada y salida del subtítulo, el texto, el estilo (de entre los descritos en la parte anterior), la posición del texto y un efecto.

Timed Text Authoring Format 1.0 – Distribution Fomat Exchange Profile (TTAF DFXP)

El TTAF DFXP [W3C, 2010 d] es una recomendación candidata de W3C. El TTAF representa, mediante texto, la información temporal asociada a un medio. Para su distribución se diseñó el DFXP, con el objetivo de intercambiar el texto con información temporal entre formatos de distribución utilizados en subtitulado.

TTAF DFXP está basado en XML, por lo que es fácil de editar y comprender. En la actualidad, es un formato que se usa, sobre todo, en reproductores Flash.

El contenido consta de:

- Etiqueta <tt> dentro de la cual se encuentra toda la información sobre los subtítulos.
- Encabezado (etiqueta <head>) en el que se especifican los diferentes estilos: tipo de fuente, tamaño, color, etc.
- Cuerpo (etiqueta <body>) donde aparece el texto de cada subtítulo, indicando el tiempo de entrada y salida del mismo y el estilo al que pertenece.

Del TTAF DFXP ha surgido el *Timed Text Markup Language* (TTML) 1.0, que se ha convertido en recomendación candidata en febrero de 2010.

2.3.3.2. Diferencias entre formatos de subtítulos

La mayoría de los formatos tienen sus ventajas y desventajas. Como regla general podemos afirmar que, cuanto más complejo es un formato, menos programas procesadores de subtítulos serán compatibles con él.

Dependiendo de las características que presenten los subtítulos pueden ser: planos, con etiquetas, con efectos y con estilos.

Formatos de subtítulos “planos”

Son los formatos más compatibles con los programas de subtitulación. Son el texto del diálogo con sus respectivos tiempos de inicio y de finalización del diálogo.

El formato de subtítulos "plano" por excelencia es el SubRip plano, siendo el formato de texto plano más utilizado.

```

1
00:07:37,927 --> 00:07:40,487
LABORATORIOS DEL FBI
WASHINGTON, D.C.

2
00:28:39,487 --> 00:28:42,445
MUELLE 39 - CENTRO DE MANDO MOVIL DEL FBI

3
00:56:56,326 --> 00:56:58,556
15 HORAS DE PLAZO

4
01:30:40,966 --> 01:30:43,844
8 HORAS DE PLAZO

```

Ejemplo 2.1. Subtítulos con el formato SubRip

Este formato ha evolucionado y ahora se le pueden agregar etiquetas (de cursiva, negrita y colores) y superponer líneas de diálogo (poner dos diálogos distintos que transcurren en el mismo lapso de tiempo). Sin embargo, hay que tener en cuenta

que esto ya no es "SubRip plano" y deja de ser compatible con programas más sencillos.

Otros formatos "planos" son: MicroDVD, SAMI, si no tiene especificaciones de colores ni de formato, TXT (.txt) y SubStation Alpha/Advanced SubStation Alpha, si no tienen ni estilos ni etiquetas.

Los formatos de subtítulos "planos" pueden convertirse fácilmente de uno a otro, sin pérdida de información durante la conversión.

El programa más usado para convertir formatos entre sí es el Subtitle Workshop, el cual es compatible con una gran variedad de formatos de subtítulos.

Formatos de subtítulos "con etiquetas"

Durante la evolución de los programas que procesan subtítulos, aparecieron versiones de los programas que podían entender lo que en informática se denomina etiquetas (*tags*). Una etiqueta marca un estilo del texto del subtítulo.

```
<i>Este subtítulo está encerrado en un tag que se llama "i".</i>
```

Ejemplo 2.2. Etiqueta para letra cursiva en el formato de subtítulos SubRip

```
{\i1}Este subtítulo está encerrado en un tag que se llama "i".{\i0}
```

Ejemplo 2.3. Etiqueta para letra cursiva en el formato de subtítulos SSA

Aunque hay distintas notaciones para las etiquetas, la característica común es que las etiquetas se dividen en dos, una *etiqueta de apertura* y una *etiqueta de cierre*, y todo lo que está encerrado en el medio, se dice que está "modificado" por esa etiqueta. En el *Ejemplo 2.2* y *Ejemplo 2.3*, la etiqueta empleada cambia el formato del texto a cursiva.

Si la etiqueta de cierre no aparece, los programas asumen que toda la línea está modificada por la misma etiqueta.

Ahora bien, no todos los programas entienden los subtítulos con etiquetas y algunos programas, lo cual es más peligroso, "creen" que los entienden, pero no los

procesan correctamente. En principio, un programa entiende de forma correcta un subtítulo con etiquetas si del *Ejemplo 2.4*, muestra en cursiva únicamente la parte del texto encerrada entre las etiquetas.

```
<i>Esta parte con cursiva.</i>Esta parte sin cursiva.
```

Ejemplo 2.4. Ejemplo de texto en cursiva correcto

Algunos programas, como el Subtitle Workshop, presentan este problema ya que si encuentran un archivo SubRip con etiquetas, lo modifica como se muestra en el *Ejemplo 2.5*.

```
<i>Esta parte con cursiva.Esta parte sin cursiva.</i>
```

Ejemplo 2.5. Ejemplo de texto en cursiva incorrecto

Formatos de subtítulos “con efectos”

Posteriormente, apareció el programa VFilter o Textsub capaz de procesar subtítulos con formato SSA/ASS con etiquetas complejas, como el *Ejemplo 2.6*, modificado por la etiqueta *move*, que mueve gradualmente un subtítulo de una posición a otra, y la etiqueta *fade*, que muestra/oculta gradualmente un subtítulo.

```
{\move(80,180,300,180)}{\fade(255,0,255,0,1000,3000,5000)} Jano
```

Ejemplo 2.6. Subtítulo con efectos

El problema del programa VFilter es que sólo puede utilizarse en sistemas Windows y modificar el código para que sea multiplataforma es una tarea complicada, dado que su código fuente está mal documentado. Por este motivo, apareció el programa LibASS, creado por el proyecto MPlayer, dando la posibilidad de ver los subtítulos sin importar la plataforma.

Formatos de subtítulos “con estilos”

Aparte de las etiquetas, cada vez más complejas, los programas empezaron a ser capaces de procesar “estilos”.

```
{\fnComic Sans MS\i1}Oia, donde quiera que vaya sólo hay bosques.  
{\fnComic Sans MS\i1}Hombre, quiero ir a Geffen y trabajar.  
{\fnComic Sans MS\i1}¿De veras? Yo quisiera continuar cazando un poco  
más.  
{\fnComic Sans MS\i1}Eso estaría bien, pero he oído que es un poco  
peligroso.  
{\fnComic Sans MS\i1}Estaremos bien. Incluso si algo ocurre,\N yo los  
protegeré con esto.
```

Ejemplo 2.7. Ejemplo de subtítulos con etiqueta para la fuente y el estilo cursiva del texto

En casos como el *Ejemplo 2.7*, lo más sencillo es crear un estilo que indique que las líneas precedidas por una misma etiqueta se modifiquen según dicha etiqueta. En este caso, el estilo indicaría que todas las líneas precedidas por la etiqueta `{\fnComic Sans MS\i1}` muestren el subtítulo con fuente Comic Sans MS y en cursiva.

La ventaja principal de utilizar estilos es que los cambios de letra, color y demás características de formato pueden realizarse una sola vez, en el estilo definido, en lugar de cambiar línea por línea.

2.4. Herramientas de subtitulado

Existen diversas herramientas o aplicaciones dedicadas a la edición y manipulación de subtítulos. La mayoría de ellas es software que se instala en el PC para poder utilizarlo, aunque en la actualidad se están desarrollando técnicas para poder subtitular de manera on-line en Internet.

En este apartado se presentarán las herramientas de subtitulado más conocidas, diferenciando entre las herramientas a instalar en local y las herramientas on-line.

2.4.1. Herramientas en PC

2.4.1.1. *Subtitle Workshop*

Es una herramienta gratuita [URUWorks, 2010] de edición de subtítulos que permite crear, editar y convertir subtítulos. Soporta la mayoría de formatos de subtítulos existentes (SubRip, MicroDVD, SubStation Alpha/Advanced SubStation Alpha, TXT, etc.) y posee una interfaz sencilla y menús de fácil acceso y con las características necesarias para realizar una subtitulación adecuada y rápida.

Esta herramienta está disponible para el sistema operativo Windows.

Las características más importantes que presenta son las siguientes:

- **Formatos e idiomas soportados:** alrededor de 56 formatos. Soporta formatos planos y con etiquetas. Permite guardar subtítulos con un formato personalizado. Además, permite crear subtítulos en varios idiomas.
- **Interfaz:** amigable, multidioma y con teclas de acceso directo configurables. Posee modo de vista previa integrado y un modo “traductor”.
- **Módulo de vídeo:** no está basado en *Windows Media Player*, lo que proporciona un rendimiento y una calidad visual mejorada. Permite crear de forma sincronizada los subtítulos con el vídeo.
- **Operaciones con el texto:** permite personalizar la fuente de los subtítulos: tipo de fuente, tamaño, estilo (negrita, cursiva y subrayado) y color. También puede añadir bordes y sombras.
- **Operaciones con los tiempos:** permite aplicar demoras o límites de duración, extensor/reductor del tiempo final de algunos subtítulos bajo algunas condiciones, definición de duraciones automáticas, posibilidad de corregir subtítulos usando los tiempos de otro subtítulo.
- **Otros aspectos:** corrector ortográfico, hacer/deshacer multinivel, posibilidad de dividir y unir subtítulos, y corrección de subtítulos: subtítulos vacíos, superposición, valores erróneos, subtítulos de más de dos líneas, duraciones demasiado largas/cortas, líneas demasiado largas, etc.

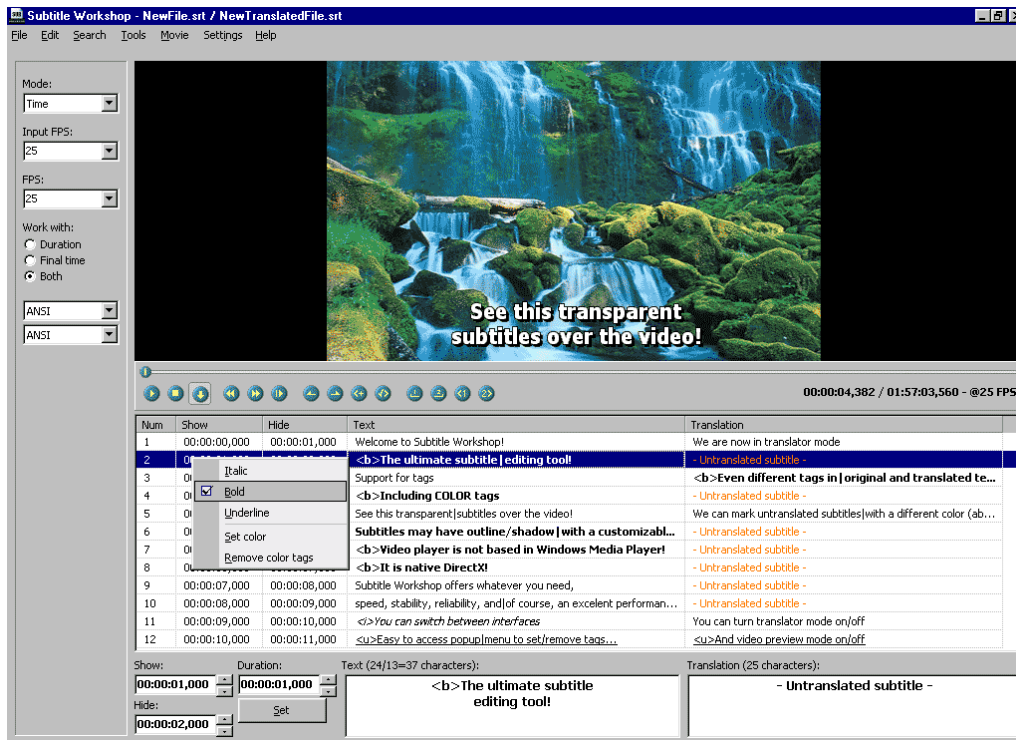


Figura 2.18. Captura de pantalla de la herramienta Subtitle Workshop

2.4.1.2. Aegisub

Es una herramienta de software gratuita de edición de subtítulos para el formato SubStation Alpha. Cuenta con una gran cantidad de herramientas convenientes que ayudan con la sincronización, composición, edición y traducción de subtítulos.

Este editor está disponible para los sistemas operativos Windows, Unix y Mac OS X.

Los aspectos más destacados de Aegisub son los siguientes [Aegisub, 2010]:

- **Formatos e idiomas soportados:** permite importación/exportación de varios formatos diferentes (Timed Text, SubRip, MicroDVD...). Además, tiene soporte completo para caracteres Unicode, con lo que permite crear subtítulos para cualquier idioma.
- **Interfaz:** simple e intuitiva. Posee teclas de acceso directo configurables. Posee modo de vista previa integrado.

- **Módulo de vídeo:** soporta la mayoría de formatos de vídeo comunes, y, además, puede usar el entorno *DirectShow*⁷ para abrir cualquier formato que pueda reproducir el reproductor de Windows Media. Permite crear de forma sincronizada los subtítulos con el vídeo.
- **Operaciones con el texto:** permite personalizar la fuente de los subtítulos: tipo de fuente, tamaño, color y estilo (negrita, cursiva y subrayado). Además, permite añadir sombras y bordes y modificar el color de éstos.
- **Operaciones con los tiempos:** se puede definir los tiempos con la interfaz de audio, utilizando el analizador de espectro o el postprocesador automático de tiempo. Éste último permite extender los tiempos inicial y/o final, mover los tiempos inicial o final para hacer continuos dos subtítulos adyacentes y comprobar si los tiempos de un subtítulo están cerca del comienzo/final de un fotograma importante para acortarlo/extenderlo a dicho fotograma.
Soporta los formatos de audio que el entorno DirectShow pueda codificar.
- **Otros aspectos:** corrección ortográfica, asistente de traducción, consejos para etiquetas no válidas, resaltado de sintaxis, hacer/deshacer multinivel, guardado automático, backup y recuperación de emergencia, división/unión de subtítulos.

⁷ DirectShow es un framework multimedia y API producidos por Microsoft para los desarrolladores de software para realizar varias operaciones con archivos o flujos multimedia.

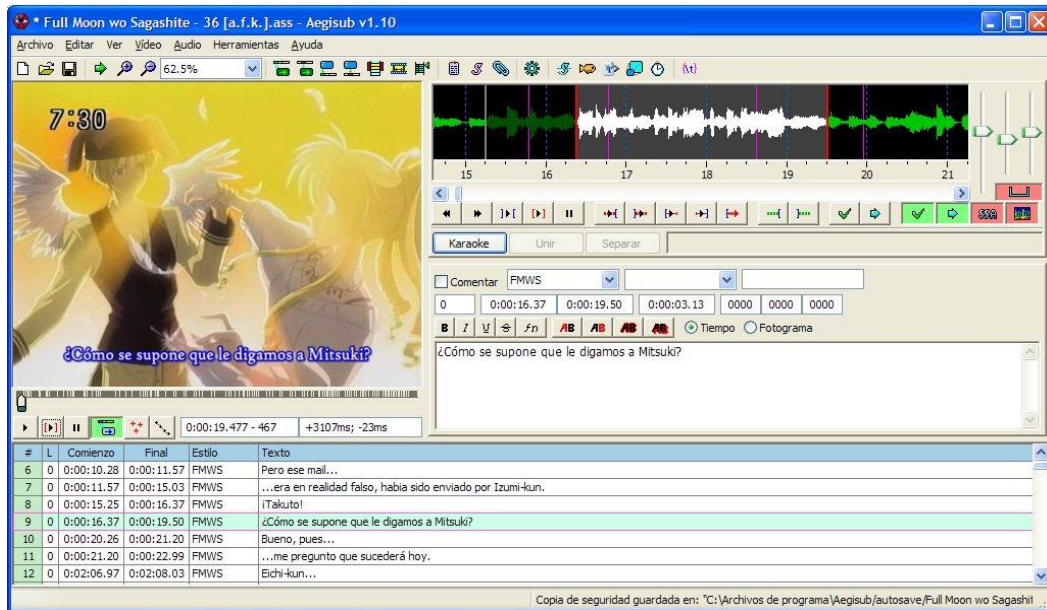


Figura 2.19. Captura de pantalla de la herramienta Aegisub

2.4.1.3. MAGpie

MAGpie [WGBH, 2010] es una aplicación gratuita desarrollada por el *National Center for Accessible Media* (NCAM). Permite añadir subtítulos, subtítulos optativos o audiodescripciones en los formatos de multimedia más utilizados: QuickTime Player, RealPlayer, Windows Media Player y Flash. En su última versión, añade compatibilidad total con Flash y el formato de subtulado DFXP.

Está disponible para los sistemas operativos Windows y Mac OS X.

Las características más importantes son [NCAM, 2009]:

- **Formatos e idiomas soportados:** permite importación/exportación de QuickTime Text y de RealText en SMIL y SAMI. Además, permite exportación a archivos DFXP para Flash y TTX⁸ para MP4 y 3GP.

Proporciona la posibilidad de crear subtítulos en varios idiomas.

- **Interfaz:** sencilla, con teclas de acceso directo configurables. Posee modo de vista previa integrado.

⁸ Formato de texto del editor *Simple Text* de Apple.

- **Módulo de vídeo:** permite crear los subtítulos de manera sincronizada con el vídeo. Incorpora los reproductores QuickTime y GRiNS (aunque en la última versión se ha eliminado GRiNS) que soportan los formatos más comunes de vídeo y audio, como .mov, .wav, .avi, .mpeg, .mp3, .mp4, etc.
- **Operaciones con el texto:** permite modificar la fuente, su tamaño, su estilo (negrita, cursiva y subrayado) y el color.
- **Operaciones con los tiempos:** no presenta.
- **Otros aspectos:** corrector ortográfico, segmentación de subtítulos, insertar subtítulos desde un archivo ya creado, control de superposición.

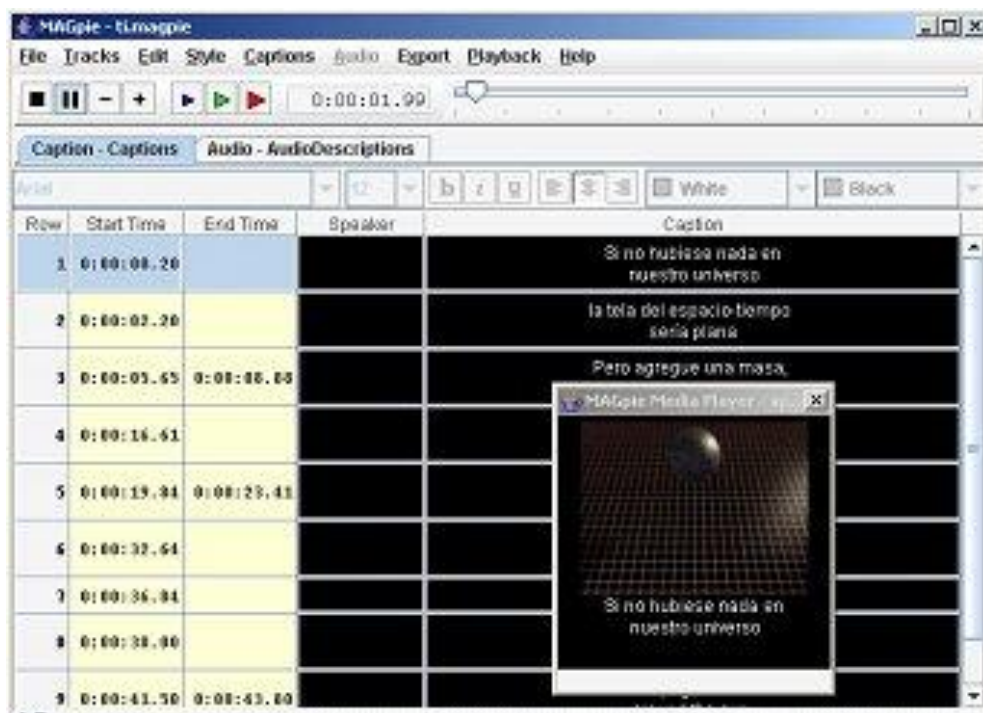


Figura 2.20. Captura de pantalla de la herramienta MAGpie

2.4.1.4. Hi-Caption Studio

Es un software comercial desarrollado por HiSoftware [HiSoftware, 2010], una empresa que también desarrolla otros aplicativos de accesibilidad. Permite crear, de forma sencilla, subtítulos para vídeo y Flash con el objetivo de ser incluido posteriormente en páginas web. Es posible crear subtítulos para vídeos SAMI, es decir,

para contenidos de Windows Media Player, aunque incorpora herramientas de conversión a formato SMIL, para RealOne Player y QuickTime.

También permite vídeos en formato Flash, permitiendo incluir el subtulado y sincronizarlo directamente desde la propia herramienta mientras se diseña la animación flash.

Está disponible sólo para Windows.

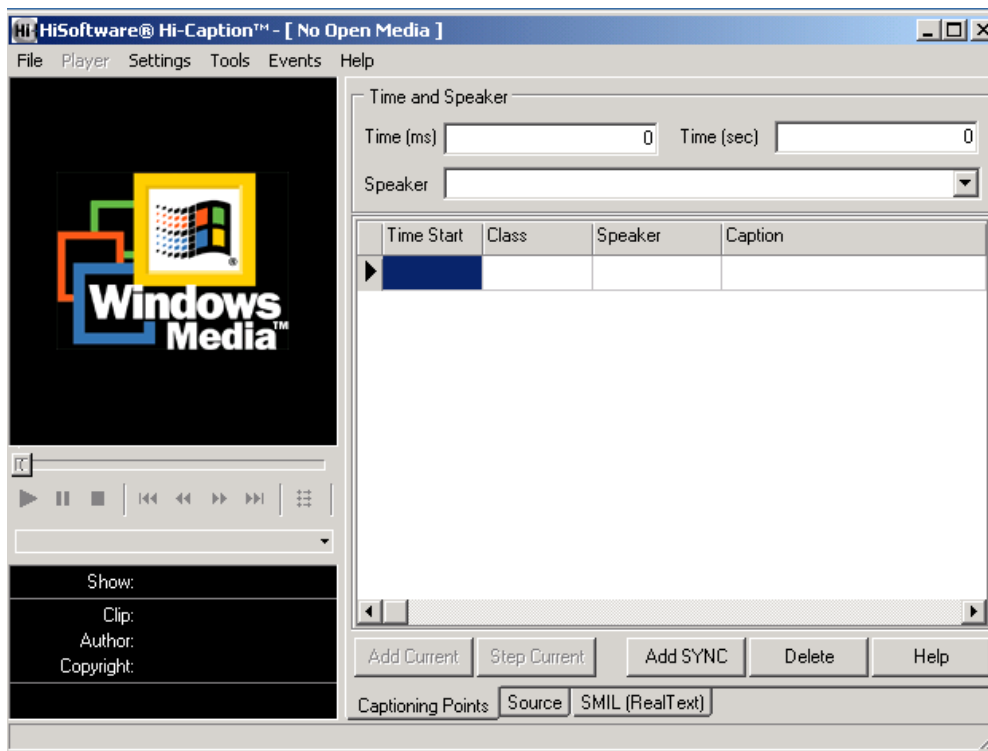


Figura 2.21. Captura de pantalla de la herramienta Hi-Caption

Las características más importantes que presenta son las siguientes [HiSoftware, 2009]:

- **Formatos e idiomas soportados:** soporta el formato SAMI y genera archivos de este formato, aunque también puede convertirlos al formato SMIL y XML. Soporta diferentes idiomas para los subtítulos.
- **Interfaz:** fácil de usar y con teclas de acceso directo configurables. Posee modo de vista previa integrado y puede personalizarse en varios idiomas.
- **Módulo de vídeo:** Windows Media Player y RealPlayer integrados en el programa. Permite crear de forma sincronizada los subtítulos con el vídeo.

- **Operaciones con el texto:** permite editar los subtítulos mediante el código SAMI, con un editor tipo HTML, o con la vista del propio programa.
- **Operaciones con los tiempos:** convierte segundos a milisegundos y viceversa para una edición manual sencilla.
- **Otros aspectos:** asistente de traducción, corrector ortográfico.

2.4.2. Herramientas on-line

2.4.2.1. Overstream

Es el editor de subtitulación on-line más conocido. El funcionamiento está basado en tecnología flash, con lo que no es necesario instalar programas ni descargar ni subir vídeos a ningún sitio.

Overstream [Overstream, 2009] es un editor sencillo e intuitivo que permite crear y sincronizar subtítulos para vídeos on-line y guardarlos en el servidor Overstream. Los vídeos a subtitular deben estar en la lista de proveedores de vídeo soportados por Overstream que, hasta el momento, son: YouTube, Google Video, MySpace Video, Dailymotion, Veoh, Blip.tv, Archive.org y Vimeo.com.

También presenta la opción *Overstream Client Services* para clientes que desean alojar sus vídeos en una ubicación específica sin que sean públicos para cualquier usuario. El cliente cuenta con su propia instancia privada del servidor Overstream en la que se puede crear un número ilimitado de usuarios.

Para utilizar un vídeo hay que introducir la URL en la que se encuentra, de donde lo tomará el reproductor de Overstream para mostrarlo y poder realizar la subtitulación de manera sincronizada, introduciendo los tiempos y el texto de cada subtítulo.

Overstream permite enviar el link del vídeo subtitulado o copiar el código HTML para embeberlo en un blog o página web. También es capaz de leer archivos de

subtítulos SubRip ya creados y de generar archivos SubRip para guardarlos en el PC. Además, soporta todos los lenguajes ya que usa la codificación UTF-8⁹.

Para usar Overstream se necesitan unos requerimientos en el navegador:

- Tener habilitados javascript y las cookies para que el sitio web funcione adecuadamente.
- Tener instalada la versión 8 o superior de Flash Player en Windows, y 9 o superior en Linux.

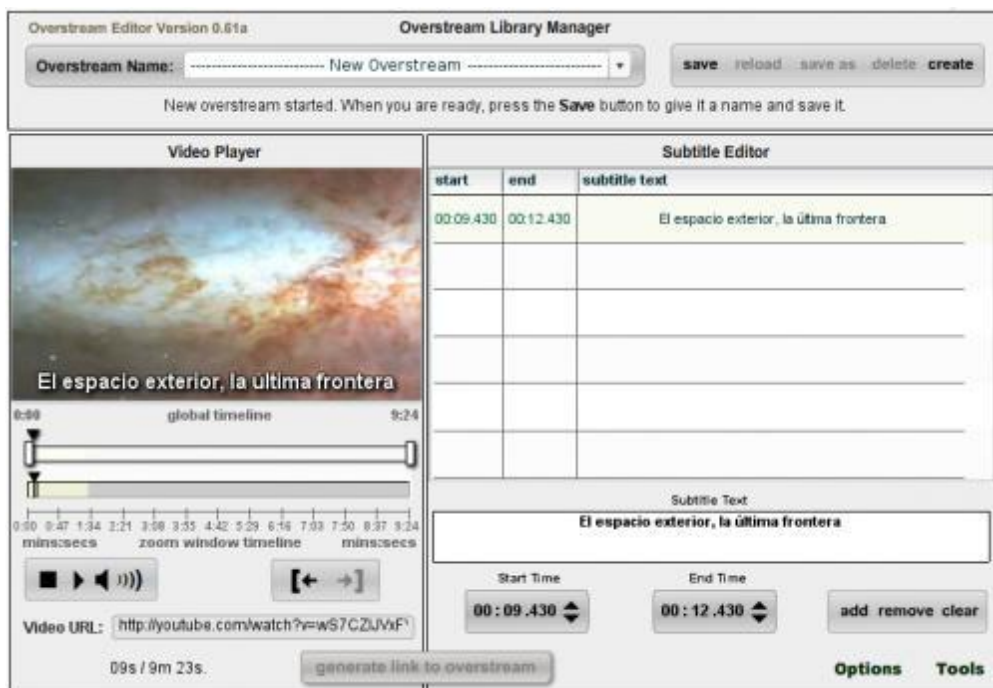


Figura 2.22. Captura de pantalla de la herramienta Overstream

2.4.2.2. CaptionTube

Existen otros editores on-line entre los que se puede destacar *CaptionTube*, para el que se necesita una cuenta de Google (Gmail) para poder acceder.

Esta herramienta tienen un funcionamiento similar a Overstream: se introduce la URL

⁹ Unicode Transformation Format 8 bit: es un formato de codificación de caracteres Unicode. Codifica cada carácter con un número variable de bytes, (de 1 a 4 bytes), dependiendo del valor entero asignado al carácter Unicode.

del vídeo, el cual se muestra en un reproductor, y se introducen los tiempos y el texto de cada subtítulo de manera sincronizada con el vídeo. Esto genera un archivo que puede guardarse en el propio PC, siempre y cuando seas el propietario del vídeo.

Esta herramienta tiene la limitación de que sólo soporta vídeos de YouTube, aunque también permite subir directamente el archivo de subtítulos al sitio web.

2.4.2.3. YouTube

El sitio web YouTube da la opción de subir archivos de subtítulos en más de 120 idiomas. Para ello, una vez subido el vídeo, se sube también el archivo, cuyo principal formato compatible es SubRip (.srt). Sin embargo, no incorpora ningún editor on-line para crear los archivos de subtítulos.

Para paliar esto, en la actualidad, ha incorporado una herramienta de reconocimiento de voz para subtítular de manera automática aunque, por el momento, la herramienta está en versión de prueba y sólo disponible en inglés.

2.4.3. Comparación de herramientas

En la *Tabla 2.2* se muestran las características más importantes en un editor de subtítulos y la evaluación de las herramientas presentadas anteriormente.

La descripción de cada característica se muestra a continuación:

- **Colores de fuente:** posibilidad de modificar el color del texto de los subtítulos.
- **Modificar formato de fuente:** opción de cambiar el formato del texto de los subtítulos, como el estilo (negrita, cursiva y subrayado), la fuente y/o el tamaño.
- **Sincronización con el vídeo:** posibilidad de crear y editar los subtítulos durante la reproducción del vídeo y de ver el resultado en tiempo real.
- **Soporte de varios idiomas:** creación de subtítulos en varios idiomas, es decir, que se permita la compatibilidad con los caracteres de los idiomas más importantes de la Unión Europea (letras como la ñ, tildes, etc).
- **Multiplataforma:** si la herramienta es compatible con varios sistemas

operativos, como Windows, Unix o Mac.

- **Control de errores:** si la herramienta controla algún tipo de error en los subtítulos, como la superposición de tiempos, etiquetas o valores erróneos, errores ortográficos, etc.
- **Gratuito:** si la herramienta puede utilizarse gratuitamente.
- **On-line:** si se puede trabajar con la herramienta sin necesidad de instalar software en el PC, a través de un portal web.
- **Importación/exportación del formato SubRip:** posibilidad de utilizar un archivo externo en formato SubRip y/o de crear un archivo en dicho formato.
- **Importación/exportación de otros formatos:** posibilidad de importar y/o exportar otros formatos, distintos de SubRip.

	Overstream	Subtitle Workshop	Aegisub	MAGpie	Hi-Caption
<i>Colores de fuente</i>		✓	✓	✓	✓
<i>Modificar formato de fuente</i>		✓	✓	✓	✓
<i>Sincronización con vídeo</i>	✓	✓	✓	✓	✓
<i>Soporte de varios idiomas</i>	✓	✓	✓	✓	✓
<i>Multiplataforma</i>	✓		✓	✓	
<i>Control de errores</i>		✓	✓	✓	✓
<i>Gratuito</i>	✓	✓	✓	✓	
<i>Online</i>	✓				
<i>Importación / exportación del formato SubRip</i>	✓	✓	✓		
<i>Importación / exportación de otros formatos</i>		✓	✓	✓	✓

Tabla 2.2. Comparativa de herramientas de subtitulación

Como se puede observar en la *Tabla 2.2*, las herramientas que precisan instalación en el PC son muy completas, ya que poseen muchas de las características importantes en una herramienta de subtítulo. Sin embargo, poseen el inconveniente de que hay que instalar la herramienta en cuestión en todos aquellos ordenadores desde los que se quiera realizar el proceso de subtitulación. Además, no todas las herramientas están disponibles para todos los sistemas operativos.

Por su parte, las herramientas on-line solucionan los problemas anteriores: pueden utilizarse sin estar instaladas en el PC y son independientes del sistema operativo, al accederse a ellas a través de Internet. No obstante, no tienen la posibilidad de dar formato al texto, como modificar el color, los estilos, la fuente, etc., ni de controlar posibles errores en los subtítulos, al contrario que las herramientas en PC, en las que estas operaciones están más desarrolladas.

2.5 UML (Unified Modeling Language)

El lenguaje de modelado unificado UML [UML, 2010] es un lenguaje estándar con el que es posible modelar todos los componentes del proceso de desarrollo de aplicaciones. Como se comenta en [Booch, 2006], UML es el lenguaje más conocido y utilizado en la actualidad y está respaldado por el OMG (*Object Management Group*) [OMG, 2010].

2.5.1. Diagrama de casos de uso

La técnica de casos de uso tiene como objetivo identificar los requisitos funcionales de un sistema estructurados en torno a las diversas categorías de usuarios. El modelo de casos de uso describe cómo un actor usa un sistema para conseguir un objetivo y lo que el sistema hace para ayudarle. Así pues, esto sirve para definir y

expresar gráficamente el sistema que nos concierne y su entorno. Gráficamente, se tiene tres elementos claramente diferenciados:

- **Casos de uso:** son las funcionalidades que contiene el sistema. Se puede definir también como una colección de escenarios (entendiendo por escenario a la secuencia de acciones del actor y las acciones del sistema que describe una interacción típica entre ambos) con un objetivo común.
- **Actores:** son los agentes externos que interaccionan con el sistema. Los actores significan roles, no entidades concretas, y más que un rol, un conjunto coherente de roles, dado que un mismo actor puede participar en varios casos de uso desempeñando un rol diferente en cada uno de ellos.
- **Asociaciones:** son las relaciones entre agentes externos y funcionalidades.

Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema, al mostrar la reacción producida como respuesta a los eventos que se producen en el mismo.

En la *Figura 2.23* se muestra un ejemplo:

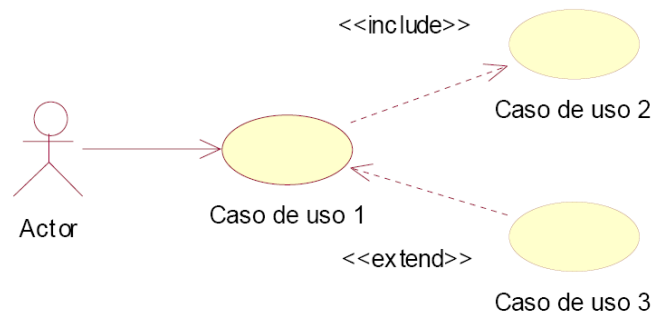


Figura 2.23. Ejemplo de casos de uso

2.5.2. Diagrama de despliegue

Los diagramas de despliegue muestran la configuración de los componentes hardware, los procesos, los elementos de procesamiento en tiempo de ejecución y los

objetos que existen en tiempo de ejecución. En este tipo de diagramas intervienen nodos, asociaciones de comunicación, componentes dentro de los nodos y objetos que se encuentran a su vez dentro de los componentes.

Un nodo es un objeto físico en tiempo de ejecución, es decir, una máquina que se compone habitualmente de, por lo menos, memoria y capacidad de procesamiento. A su vez puede estar formado por otros componentes.

En la *Figura 2.24* se puede ver un ejemplo de diagrama de despliegue:

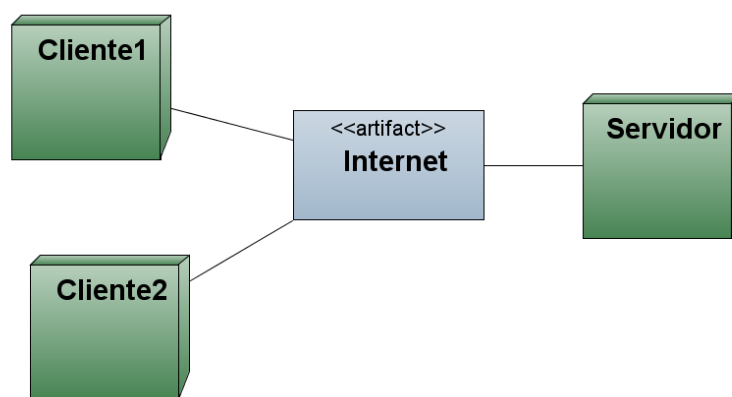


Figura 2.24. Ejemplo de diagrama de despliegue

2.5.3. Diagrama de clases

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones.

Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. Principalmente, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones y modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los de despliegue.

Algunos de los elementos que se pueden clasificar como estáticos en UML, son los siguientes:

- **Paquete:** Es el mecanismo de que dispone UML para organizar sus elementos en grupos.

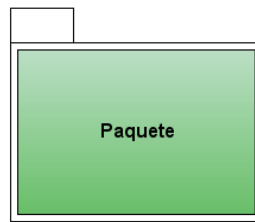


Figura 2.25. Representación gráfica de un paquete

- **Clase:** Describe un conjunto de objetos que comparte los mismos atributos, operaciones, métodos, relaciones y significado. Los componentes de una clase son:
 - **Atributo:** Se corresponde con las propiedades de una clase.
 - **Operación:** También conocido como método, es un servicio proporcionado por la clase que puede ser solicitado por otras clases y que produce un comportamiento en ellas cuando se realiza.

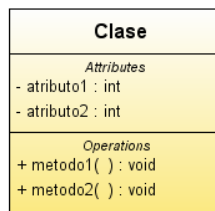


Figura 2.26. Representación gráfica de una clase

- **Interfaz:** Representa a la funcionalidad que proporciona uno o varios elementos del modelo al resto de elementos. Esta interfaz no tiene asociado un comportamiento, por lo que tendrá que ser implementado por otro elemento. En estos diagramas se muestra, además, las relaciones estructurales existentes entre los elementos descritos anteriormente.

2.5.4. Diagramas de estados

Representan la secuencia de los posibles estados por los que un objeto, o una interacción entre objetos, transcurre durante su tiempo de vida, en respuesta a los eventos recibidos. Modela el sistema como una máquina de estados.

Un estado en UML, se puede definir como la situación en que se encuentra un objeto o una interacción cuando satisface una condición, desarrolla alguna acción o se encuentra esperando un evento. Cuando un objeto o una interacción pasa de un estado a otro, debido a un evento, se dice que ha sufrido una transición. Puede suceder, que con la aparición de un evento, el estado de origen y el de destino sea el mismo, teniendo así, una transición reflexiva.

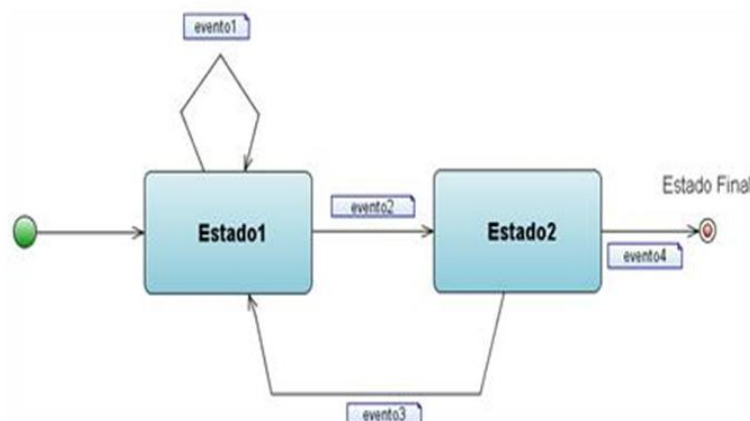


Figura 2.27. Ejemplo diagrama de estados

2.5.5. Diagrama de actividades

Son similares a los diagramas de flujo tradicionales. En realidad, se corresponden con un caso especial de los diagramas de estado, donde los estados son estados de acción, y las transiciones vienen provocadas por la finalización de las acciones que tienen lugar en los estados de origen. Es decir, estados con una acción interna y una o más transiciones que suceden al finalizar esta acción, o dicho de otro modo, un paso en la ejecución de lo que será un procedimiento.

Generalmente, estos diagramas están asociados a la implementación de un caso de uso, o de un método. Los diagramas de actividad se utilizan para mostrar el flujo de operaciones que se desencadenan en un procedimiento interno del sistema.

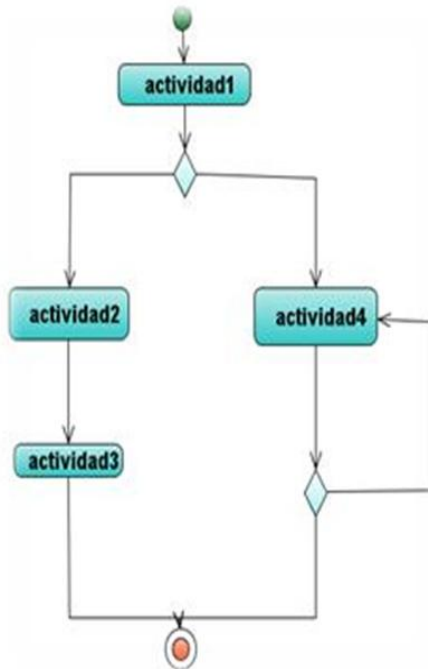


Figura 2.28. Ejemplo diagrama de actividades

CAPÍTULO 3. HERRAMIENTAS PARA LA ELABORACIÓN DEL PROYECTO

En este apartado se incluyen las herramientas empleadas para realizar el proyecto. En primer lugar, se detallarán las características del servidor en el que se encuentra la aplicación y, en el siguiente apartado, las tecnologías utilizadas en el desarrollo de la misma.

3.1. Infraestructura del Servidor

Respecto a la infraestructura del servidor, se ha empleado la desplegada para el Gestor de contenido multimedia de la Universidad Carlos III de Madrid, debido a que la aplicación debe estar integrada en dicho Gestor. De esta forma, también se obtiene un mejor aprovechamiento de recursos y una mayor seguridad.

La infraestructura puede resumirse en un servidor con las siguientes características:

- HP Compaq Proliant 330
- Procesador Pentium IV a 1,5GHz
- Memoria RAM de 1GB
- Disco duro de 150GB
- Sistema operativo Linux Debian Server 4.0

3.1.1. Servidor Web

Un Servidor Web [Fernández-Tostado, 2009] es un programa que implementa el protocolo HTTP (*HyperText Transfer Protocol*). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (*HyperText Markup Language*): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.

Es importante destacar que comúnmente se utiliza la palabra servidor para referirnos a un ordenador con un software servidor instalado, pero un servidor también es el software que permite la realización de las funciones descritas en el párrafo anterior.

3.1.1.1. Arquitectura cliente/servidor

Para explicar la labor de un servidor web, se parte de la arquitectura cliente/servidor. El funcionamiento de este modelo consiste en que el cliente (un navegador web) que quiere llevar a cabo una operación, se la traslada al servidor, mediante una petición HTTP o HTTPS (la versión cifrada y autenticada), que se encarga de realizarla y de devolverle el resultado, el cual se muestra en el navegador del cliente como página web.

El servidor web puede transferir código HTML o enviar aplicaciones web. Estas aplicaciones son porciones de código que se ejecutan cuando se realizan ciertas peticiones o respuestas HTTP. Se distingue entre:

- **Aplicaciones en el lado del cliente:** el cliente web se encarga de ejecutar estas aplicaciones en la máquina del usuario. Son las aplicaciones tipo Java "applets" o Javascript: el servidor proporciona el código de las aplicaciones al cliente y éste, mediante el navegador, las ejecuta. Es necesario, por tanto, que el cliente disponga de un navegador con capacidad para ejecutar aplicaciones (también llamadas scripts). Comúnmente, los navegadores permiten ejecutar aplicaciones escritas en lenguaje javascript y java, aunque pueden añadirse más lenguajes mediante el uso de plugins.

- **Aplicaciones en el lado del servidor:** el servidor web ejecuta la aplicación; ésta, una vez ejecutada, genera cierto código HTML; el servidor toma este código recién creado y lo envía al cliente por medio del protocolo HTTP (o HTTPS).



Figura 3.1. Arquitectura cliente/servidor

La mejor opción para realizar aplicaciones web suele ser las aplicaciones de servidor, ya que, al no ejecutarse la aplicación en la máquina del cliente, éste no precisa de ninguna capacidad añadida.

Otro dato a destacar es que una misma máquina puede ofrecer unos servicios a la red y usar los que sirven otros sistemas, es decir, con el rol de servidor o de cliente según sea necesario.

3.1.1.2. Servidor web Apache

Gracias a los avances en conectividad y la gran disponibilidad de banda ancha, ya no es necesario recurrir a caros alojamientos en proveedores externos para establecer un servidor web. Esto es posible gracias a Apache, uno de los servidores web más utilizados debido a las características que presenta.



Figura 3.2. Logo de Apache

En este proyecto, Apache es el servidor utilizado en la máquina donde se aloja la aplicación desarrollada.

Apache [Apache, 2009] es un servidor web HTTP de código abierto multiplataforma. Es un servidor seguro, eficiente y rápido, que está continuamente actualizado y adaptado a los nuevos protocolos.

El servidor Apache se desarrolla dentro del proyecto *HTTP Server* de *Apache Software Foundation*. Entre sus características destacan [Andrés, 2009]:

- **Multiplataforma.** Funciona en la mayoría de las plataformas actuales. Gracias a esta característica, se da libertad al programador para que elija la plataforma que más se adapte a sus necesidades.
- **Modular.** Apache puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona. Esta arquitectura modular permite construir un servidor hecho a medida. Además permite la implementación de los últimos y más nuevos protocolos.
- **Extensible.** Apache se puede extender, lo que permite construir módulos que le den nuevas funcionalidades, que son cargadas en tiempos de ejecución.
- **Código abierto.** Esta característica tiene la ventaja de que el código fuente esté disponible, haciendo posible que usuarios, programadores, y empresas puedan involucrarse en el desarrollo de las aplicaciones y, de esta forma, el proceso de corrección y detección de errores se lleve a cabo de forma eficiente.

3.1.2. Sistema de Gestión de Bases de Datos

El objetivo general de los Sistemas de Gestión de Base de Datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, asegurando su integridad, confidencialidad y seguridad.

El Sistema Gestor de Bases de Datos que se emplea en el servidor de este proyecto es MySQL, del cual se hablará en el siguiente apartado.

3.1.2.1. MySQL

MySQL [Pérez, 2007] es desarrollado y mantenido por MySQL AB, una subsidiaria de *Sun Microsystems*, desde enero de 2008, y ésta a su vez de *Oracle Corporation*, desde abril de 2009. Es un sistema de gestión de base de datos relacional, multihilo y multiusuario.



Figura 3.3. Logo de MySQL

MySQL destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados, como PHP, Perl y Java, y su integración en distintos sistemas operativos.

También es muy destacable la condición de “código libre” que hace que su utilización sea gratuita e, incluso, que se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido positivamente en su desarrollo y continuas actualizaciones para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet.

Las principales características que presenta este gestor son:

- **Velocidad.** Es rápido.
- **Facilidad de uso.** Es un sistema de base de datos de alto rendimiento pero relativamente simple y es mucho menos complejo de configurar y administrar que sistemas más grandes.
- **Coste.** Es gratuito.

- **Capacidad de gestión de lenguajes de consulta.** MySQL comprende SQL, el lenguaje elegido para todos los sistemas de bases de datos modernos.
- **Capacidad.** Pueden conectarse muchos clientes simultáneamente al servidor. Los clientes pueden utilizar varias bases de datos simultáneamente. Además, está disponible una amplia variedad de interfaces de programación para lenguajes como C, Perl, Java, PHP y Python.
- **Conectividad y seguridad.** MySQL está completamente preparado para el trabajo en red y las bases de datos pueden ser accedidas desde cualquier lugar de Internet. Dispone de control de acceso.
- **Portabilidad.** MySQL se puede utilizar en una gran cantidad de sistemas Unix diferentes, así como bajo Microsoft Windows.
- **Distribución abierta.** Puede obtener y modificar el código fuente de MySQL.

3.2. Entorno de trabajo

En este apartado se detalla los lenguajes utilizados para el desarrollo de la aplicación y sus características más importantes.

3.2.1. Adobe Flash y ActionScript

3.2.1.1. Adobe Flash

Adobe Flash [Adobe, 2010 b] es una aplicación privativa destinada a la producción y entrega de contenido interactivo para ordenadores personales, dispositivos móviles y pantallas de prácticamente cualquier tamaño y resolución. Se diferencia de otras formas de animación (como gif, Power Point, etc.) en que puede tener interactividad con el usuario e interconexión con otros códigos.



Figura 3.4. Logo de Adobe Flash

No hay que confundir Flash con *Flash Player*. El primero [Adobe, 2010 a] es el entorno para el desarrollo de contenido interactivo, además de interfaces de usuario y aplicaciones web. El segundo, el programa de máquina virtual que muestra el contenido creado con Flash.

Los archivos de Flash, de extensión SWF (*Small Web Format*), proporcionan gráficos vectoriales, imágenes de mapa de bits, texto, vídeo, sonido y audio a través de Internet. Suelen aparecer como interfaces de aplicaciones web, interfaces de usuario, animaciones o publicidad interactiva.

Clips de Película en Adobe Flash

La clase *MovieClip* [Adobe, 2007] es la clase principal para animación y símbolos de clip de película creados en *Adobe Flash*. Tiene todos los comportamientos y la funcionalidad de los objetos de visualización, pero con propiedades y métodos adicionales para controlar la línea de tiempo de un clip de película.

Los clips de película son un elemento clave para las personas que crean contenido animado con la herramienta de edición Flash y desean controlar el contenido con *ActionScript* (se hablará posteriormente de él). Siempre que se crea un símbolo de clip de película en Flash, Flash añade el símbolo a la biblioteca del documento Flash en el que se crea. De manera predeterminada, este símbolo se convierte en una instancia de la clase *MovieClip*, y como tal tiene las propiedades y los métodos de esta clase.

Cuando se coloca una instancia de un símbolo de clip de película en el escenario, el clip de película progresa automáticamente por su línea de tiempo. Esta línea de tiempo es lo que distingue a la clase *MovieClip*, que permite crear animación mediante interpolaciones de movimiento y de forma, a través de la herramienta de edición de Flash.

3.2.1.2. *ActionScript*

ActionScript [Adobe, 2007] es el lenguaje de programación orientado a objetos para el entorno de tiempo de ejecución de *Adobe Flash Player*. Activa, entre otras muchas cosas, la interactividad y la gestión de datos en el contenido y las aplicaciones de Flash.

ActionScript se ejecuta mediante la máquina virtual ActionScript (AVM), que forma parte de *Flash Player*. El código ActionScript se suele compilar a un formato de código de bytes (un tipo de lenguaje que los ordenadores pueden escribir y comprender) mediante un compilador, como el incorporado en *Adobe Flash* o en *Adobe Flex Builder*, o el que está disponible en el SDK de *Adobe Flex* y en *Flex Data Services*. El código de bytes está incorporado en los archivos SWF ejecutados por *Flash Player*, el entorno de tiempo de ejecución.

La última versión de ActionScript y la más extendida actualmente es la 3.0. Algunas de las principales funciones de ActionScript 3.0 son:

- Una nueva máquina virtual ActionScript, denominada AVM2, que utiliza un nuevo conjunto de instrucciones de código de bytes y proporciona importantes mejoras de rendimiento.
- Una base de código de compilador más moderna, que se ajusta mejor al estándar ECMAScript (ECMA 262) y que realiza mejores optimizaciones que las versiones anteriores del compilador.
- Una interfaz de programación de aplicaciones (API) ampliada y mejorada, con un control de bajo nivel de los objetos y un auténtico modelo orientado a objetos.

- Un núcleo del lenguaje basado en el próximo borrador de especificación del lenguaje ECMAScript (ECMA-262) edición 4.
- Una API XML basada en la especificación de ECMAScript para XML (E4X) (ECMA-357 edición 2). E4X es una extensión del lenguaje ECMAScript que añade XML como un tipo de datos nativo del lenguaje.
- Un modelo de eventos basado en la especificación de eventos DOM (modelo de objetos de documento) de nivel 3.

Ventajas de ActionScript 3.0

ActionScript 3.0 [Adobe, 2007] aumenta las posibilidades de creación de scripts de las versiones anteriores de ActionScript. Se ha diseñado para facilitar la creación de aplicaciones muy complejas con conjuntos de datos voluminosos y bases de código reutilizables y orientadas a objetos. También, ActionScript 3.0 permite introducir unas mejoras de rendimiento que sólo están disponibles con AVM2, la nueva máquina virtual. El código ActionScript 3.0 puede ejecutarse con una velocidad diez veces mayor que el código ActionScript heredado.

3.2.2. PHP (Hipertext Preprocessor)

PHP [PHP, 2010] es un lenguaje de programación interpretado de código abierto especialmente adecuado para desarrollo web y que puede ser incrustado en HTML, es decir, se incluyen scripts dentro del código HTML.

El código PHP se ejecuta en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá los resultados de ejecutar el script, sin ninguna posibilidad de determinar qué código ha producido el resultado recibido.



Figura 3.5. Logo de PHP

PHP puede realizar tareas como procesar la información de formularios, generar páginas con contenidos dinámicos o enviar y recibir cookies.

Los tres campos principales en los que se usan scripts en PHP son:

- **Scripts del lado del servidor.** Este es el campo más tradicional y el principal foco de trabajo. Se necesitan tres cosas para que esto funcione. El intérprete PHP (CGI módulo), un servidor web y un navegador. Es necesario hacer funcionar el servidor, con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador del cliente, conectándose con el servidor web.
- **Scripts en la línea de comandos.** Puede crear un script PHP y correrlo sin necesidad de un servidor web o navegador. Solamente necesita el intérprete PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en *nix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesamiento de texto.
- **Escribir aplicaciones de interfaz gráfica.** Probablemente PHP no sea el lenguaje más apropiado para escribir aplicaciones gráficas, pero para utilizar algunas características avanzadas en programas clientes, se puede emplear PHP-GTK para escribir dichos programas. También es posible escribir aplicaciones independientes de una plataforma. PHP-GTK es una extensión de PHP, no disponible en la distribución principal.

Además, PHP tiene la ventaja de que puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, interactuar con la mayoría de servidores web de hoy en día y permitir la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, etc.

En este trabajo, se ha utilizado la versión 5 de este lenguaje debido a las ventajas que posee.

3.2.2.1. PHP 5

Con las primeras dos versiones de PHP [Álvarez, 2004], PHP3 y PHP4, se había conseguido una plataforma potente y estable para la programación de páginas del lado del servidor. Sin embargo, existían puntos negros en el desarrollo PHP que se han tratado de solucionar con la versión 5.

El principal objetivo de PHP5, lanzada en julio de 2004, ha sido mejorar los mecanismos de la programación orientada a objetos, ya que en las anteriores versiones sólo se implementaba una parte muy pequeña de las características de este tipo de programación.

Algunas de las características que incluye PHP5 son [Herrera, 2007]:

- Mejor soporte para la programación orientada a objetos.
- Mejor soporte para XML, con librerías más potentes y más estándares.
- Mejor soporte para MySQL, con una extensión completamente reescrita, MySQLi.
- Aumento de las ventajas sobre la extensión SQLite.
- Soporte para la librería Tidy, poderosa para parsear, entre otras muchas cosas, en documentos HTML.
- Nuevo administrador de memoria que libera los bloques de memoria con mucha más rapidez y eficiencia.

3.2.3. JavaScript

JavaScript es un lenguaje de programación orientado a objetos utilizado para acceder a objetos en aplicaciones. Dichos objetos facilitan la programación de páginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la máquina del usuario, tales como formateo de unidades, modificar archivos, etc.

Principalmente, [Eguíluz, 2009] se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas. Sus utilidades más importantes son:

- Páginas web dinámicas: creación de galerías de imágenes, creación de mensajes *Tooltip*, etc.
- Validación de datos en formularios.
- Posibilidad de ejecutar código de manera no secuencial.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, es un lenguaje que funciona del lado del cliente ya que los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

Una característica importante de JavaScript reside en que es compatible con cualquier navegador y sistema operativo, lo que le convierte en el lenguaje de programación del lado del cliente más empleado, frente a otros lenguajes, como Visual Basic Script, desarrollado por Microsoft, que sólo puede utilizarse en el navegador Internet Explorer.

3.2.3.1. Posibilidades y limitaciones de JavaScript

Desde su aparición, [Eguíluz, 2009] JavaScript siempre fue utilizado de forma masiva por la mayoría de sitios de Internet. La aparición de Flash disminuyó su popularidad, ya que Flash permitía realizar algunas acciones imposibles de llevar a cabo mediante JavaScript.

Sin embargo, la aparición de las aplicaciones AJAX programadas con JavaScript le ha devuelto una mayor popularidad dentro de los lenguajes de programación web.

En cuanto a las limitaciones, JavaScript fue diseñado de forma que se ejecutara en un entorno muy limitado que permitiera a los usuarios confiar en la ejecución de los scripts.

De esta forma, los scripts de JavaScript no pueden comunicarse con recursos que no pertenezcan al mismo dominio desde el que se descargó el script. Los scripts tampoco pueden cerrar ventanas que no hayan abierto esos mismos scripts. Las ventanas que se crean no pueden ser demasiado pequeñas ni demasiado grandes ni colocarse fuera de la vista del usuario (aunque los detalles concretos dependen de cada navegador).

Además, los scripts no pueden acceder a los archivos del ordenador del usuario (ni en modo lectura ni en modo escritura) y tampoco pueden leer o modificar las preferencias del navegador.

Por último, si la ejecución de un script dura demasiado tiempo (por ejemplo por un error de programación) el navegador informa al usuario de que un script está consumiendo demasiados recursos y le da la posibilidad de detener su ejecución.

A pesar de todo, existen alternativas para poder saltarse algunas de las limitaciones anteriores. La alternativa más utilizada y conocida consiste en firmar digitalmente el script y solicitar al usuario el permiso para realizar esas acciones.

3.2.4. XML (eXtensible Markup Language)

XML [W3C, 2010 a] es un lenguaje de etiquetado extensible desarrollado por el *World Wide Web Consortium* (W3C). Es una simplificación y adaptación del SGML y su función principal es describir datos y no mostrarlos como es el caso de, por ejemplo, HTML. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones.

Por tanto, no es realmente un lenguaje, sino una manera de definir lenguajes para diferentes necesidades. Así, XML sirve para estructurar, almacenar e intercambiar información.

3.2.4.1. Ventajas de XML

Las ventajas más importantes que presenta este lenguaje son:

- Es **extensible**: después de diseñado y puesto en producción, es posible extender XML con la adición de nuevas etiquetas, de modo que se pueda continuar utilizando sin complicación alguna.
- El **analizador** es un **componente estándar**, no es necesario crear un analizador específico para cada versión de lenguaje XML. Esto posibilita el empleo de cualquiera de los analizadores disponibles. De esta manera se evitan *bugs* y se acelera el desarrollo de aplicaciones.
- Si un tercero decide usar un documento creado en XML, es **sencillo** entender su estructura y procesarla. Mejora la compatibilidad entre aplicaciones.

3.2.4.2. Documentos XML bien formados

Los documentos denominados como “bien formados” (del inglés *well formed*) son aquellos que cumplen con todas las definiciones básicas de formato y pueden, por lo tanto, analizarse correctamente por cualquier analizador sintáctico (*parser*) que cumpla con la norma. A continuación, [Reino, 2007] se detallan las características necesarias para un documento bien formado:

- Los documentos han de seguir una **estructura estrictamente jerárquica** con lo que respecta a las etiquetas que delimitan sus elementos. Una etiqueta debe estar correctamente incluida en otra, es decir, las etiquetas deben estar correctamente anidadas. Además, los elementos con contenido deben estar correctamente cerrados.
- Los documentos XML sólo permiten **un elemento raíz** del que todos los demás sean parte, es decir, solo pueden tener un elemento inicial.
- Los valores de los **atributos** siempre deben estar encerrados entre **comillas simples o dobles**.
- XML es **sensible a mayúsculas y minúsculas**, es decir, las etiquetas de principio y final deben estar escritas de la misma manera.
- Existe un conjunto de **caracteres** llamados “**espacios en blanco**” que los procesadores XML no ignoran, a diferencia de HTML que convierte a un sólo espacio. Estos caracteres son: espacios, tabuladores, retornos de carro y saltos de línea.

- Es necesario **asignar nombres** a las estructuras, tipos de elementos, entidades, elementos particulares, etc. En XML los nombres tienen alguna característica en común.
- Las construcciones como etiquetas, referencias de entidad y declaraciones se denominan “**marcas**”. Éstas son las **partes del documento que el procesador XML espera entender**. El resto del documento entre marcas son los datos “entendibles” por las personas.

3.2.4.3. Validez de un documento XML

Un documento XML “válido” es un documento XML “bien formado” que también se ajusta a unas normas que definen la relación que debe verificarse entre los distintos elementos presentes en el documento.

Esta relación entre elementos se especifica en un documento, que puede ser externo o estar incluido en el propio documento XML, y que está expresado como *Document Type Definition* (DTD) [W3C, 2006 a] o como *XML Schema Definition* (XSD) [W3C, 2010f]:

- **Document Type Definition**: define los tipos de elementos, atributos y entidades permitidas y puede expresar algunas limitaciones para combinarlos.
- **XML Schemas Definition**: es similar a un DTD. Define qué elementos puede contener un documento XML, cómo están organizados y qué atributos y de qué tipo pueden tener sus elementos. Presenta ciertas ventajas sobre un DTD:
 - Usa sintaxis de XML, al contrario que el DTD.
 - Permite especificar los tipos de datos.
 - Es extensible.

3.2.4.4. Tecnologías XML

Entre las tecnologías XML disponibles [W3C, 2008 c] se puede destacar:

- **XSL** [W3C, 2006 b]: Lenguaje Extensible de Hojas de Estilo, cuyo objetivo principal es mostrar cómo debería estar estructurado el contenido, cómo debería ser diseñado el contenido de origen y cómo debería ser paginado en un

medio de presentación como puede ser una ventana de un navegador web o un dispositivo móvil, o un conjunto de páginas de un catálogo, informe o libro.

- **XPath** [W3C, 2007 a]: Lenguaje de Rutas XML, es un lenguaje para acceder a partes de un documento XML.
- **XLink** [W3C, 2010 e]: Lenguaje de Enlace XML, es un lenguaje que permite insertar elementos en documentos XML para crear enlaces entre recursos XML.
- **XPointer** [W3C, 2002]: Lenguaje de Direccionamiento XML, es un lenguaje que permite el acceso a la estructura interna de un documento XML, esto es, a sus elementos, atributos y contenido.
- **XQL** [W3C, 2007 b]: Lenguaje de Consulta XML, es un lenguaje que facilita la extracción de datos desde documentos XML. Ofrece la posibilidad de realizar consultas flexibles para extraer datos de documentos XML en la Web.

3.2.5. EditPlus

Es el programa que se ha utilizado para desarrollar la aplicación propuesta en este proyecto mediante los lenguajes de programación comentados anteriormente.

EditPlus [Edit+, 2010] es un editor de texto, HTML, PHP y Java para el sistema operativo Windows. Este editor no sólo puede servir como reemplazo para el bloc de notas, sino que también ofrece potentes funciones para autores y programadores de páginas web.

EditPlus ofrece las siguientes características:

- **Resaltador de sintaxis:** permite el realce de sintaxis para los lenguajes HTML, CSS, PHP, ASP, Perl, C/C++, Java, JavaScript y VBScript, por defecto. También, permite crear un archivo de sintaxis propio para soportar otros lenguajes de programación.

- **Características de Internet:**
 - Navegador web para previsualizar el contenido de documentos HTML o applets de Java sin dejar el editor. La ventana del navegador también contiene los comandos comunes de navegadores que permiten la navegación por sitios web y por archivos de HTML en local.
 - Comandos FTP para subir los archivos locales a un servidor FTP o para editar los archivos en remoto directamente.
 - Resaltador de URL y direcciones de correo electrónico en archivos de texto que pueden activarse pulsando la tecla F8 o con “Ctrl+doble clic”.
- **Barra de herramientas de HTML:** permite insertar las etiquetas más comunes rápida y fácilmente.
- **Herramientas de usuario:** EditPlus soporta herramientas definidas por el usuario, archivos de ayuda y archivos de grabación de las teclas presionadas. La salida de la ejecución de la herramienta puede capturarse en el *Output Window* para poder seleccionar la línea de error con doble clic y cargar automáticamente el archivo y posicionar el ratón en esa línea.
- **Auto-completar:** es una herramienta que sustituye una abreviación por una cadena completa. Soporta Perl y C/C++ por defecto, pero también permite crear un archivo para soportar otros lenguajes de programación.
- **Portapapeles:** permite acceder a clips de texto de manera rápida y sencilla. Además, permite personalizarlo fácilmente y crear un archivo propio de portapapeles.
- **Otras características:** número de línea, regla, búsqueda y reemplazo de palabras, rehacer y deshacer, corrector ortográfico y atajos de teclado personalizables.
- **Codificaciones:** ASCII, UTF-8, UTF-16 y cualquier codificación soportada por el Sistema Operativo.

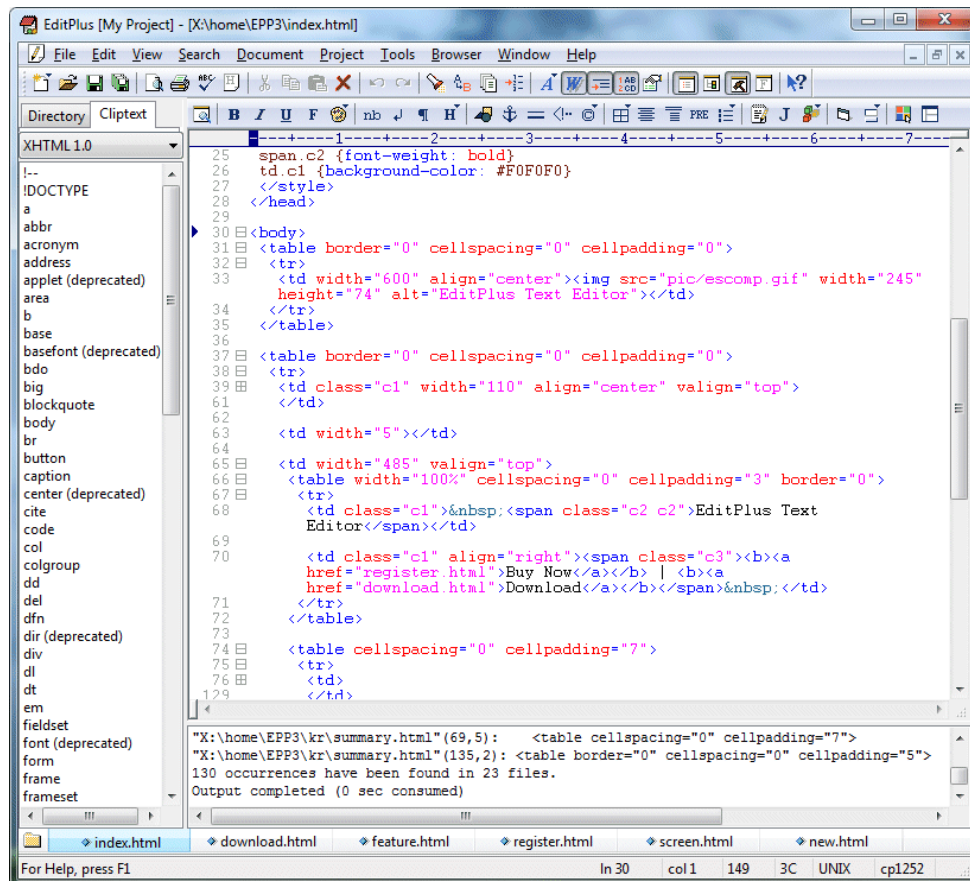


Figura 3.6. Captura de pantalla de la herramienta EditPlus

3.3. Librerías

Para crear el reproductor se han utilizado unas clases ya definidas por Flash. Las clases utilizadas son las siguientes:

Clase Video

La clase Video muestra vídeo grabado o en vivo, en una aplicación, sin incorporar el vídeo al archivo SWF. Esta clase crea un objeto Video en una interfaz de *Adobe Flash* o *Adobe Flex* que reproduce cualquiera de los siguientes tipos de vídeo:

- Archivos Flash Video (FLV) grabados y almacenados en un servidor o localmente.
- Vídeo en vivo capturado del equipo de un usuario.

Un objeto Video es un objeto de visualización y representa el espacio visual en el que se ejecuta el vídeo en una interfaz de usuario.

Cuando se utiliza con *Flash Media Server*, el objeto Video permite enviar vídeo en vivo desde el equipo de un usuario al servidor y después difundirlo desde el servidor a los usuarios. Mediante el uso de estas funciones, se pueden desarrollar aplicaciones como: un sencillo reproductor de vídeo, un reproductor de vídeo con multipunto que publica de un servidor a otro, o una aplicación de vídeo compartida para una comunidad de usuarios.

Además, se puede controlar diversas propiedades de los objetos Video. Por ejemplo, se puede desplazar el objeto Video por el escenario utilizando sus propiedades *x* (*eje de abscisas*) e *y* (*eje de coordenadas*), cambiarle el tamaño mediante sus propiedades *height* y *width*, etc.

Clase Button

El componente *Button* representa un botón rectangular de uso frecuente. Los componentes *Button* muestran una etiqueta de texto, un icono o ambos elementos.

Los componentes *Button* se suelen asociar a un método de controlador de eventos que detecta un evento *clic* y realiza la tarea especificada tras distribuir el evento. Si el usuario hace clic en un botón activado, éste distribuye los eventos *clic* y *buttonDown*. Incluso si no está activado, un botón distribuye otros eventos, entre los que se incluyen *mouseMove*, *mouseOver*, *mouseOut*, *rollOver*, *rollOut*, *mouseDown* y *mouseUp*.

La apariencia del botón se puede cambiar asociando un aspecto diferente a cada estado del botón. Los componentes *Button* también se pueden definir para que funcionen como botones de comando o de activación.

Propiedades de la clase FLVPlayback

La clase *FLVPlayback* permite incluir un reproductor de vídeo en la aplicación Flash para reproducir archivos de vídeo Flash (FLV). Estos archivos pueden ser descargados de forma progresiva, a través de http, o reproducidos sin interrupción, desde *Flash Media Server* (FMS) o desde *Flash Video Streaming Service* (FVSS).

Las propiedades utilizadas son:

- *MuteButton*: Control del botón de silencio.
- *PlayPauseButton*: Control del botón de reproducción/pausa.
- *SeekBar*: Control de la barra de reproducción/carga.
- *StopButton*: Control del botón de detención.
- *VolumeBar*: Control de la barra de volumen.

Propiedades de la clase FLVPlaybackCaptioning

El componente *FLVPlaybackCaptioning* activa el texto de *FLVPlayback*. El componente *FLVPlaybackCaptioning* descarga un archivo XML de texto temporizado (TT) y aplica dicho texto a un componente *FLVPlayback* con el que *FLVPlaybackCaptioning* está asociado.

La propiedad empleada es:

- *CaptionButton*: Define la instancia del componente *CaptionButton* de la interfaz personalizada *FLVPlayback*, que proporciona capacidades de conmutación para activar y desactivar el texto. La propiedad *captionButton* funciona de forma similar a las propiedades de *FLVPlayback* *playButton*, *pauseButton*, *muteButton*, etc.

Clase NetConnection

La clase *NetConnection* crea una conexión bidireccional entre *Flash Player* y una aplicación *Flash Media Server*, o bien entre *Flash Player* y un servidor de aplicaciones que ejecute *Flash Remoting*. El objeto *NetConnection* es como un canal entre el cliente y el servidor.

Sin *Flash Media Server*, se puede utilizar la clase *NetConnection* para reproducir archivos de vídeo y MP3 desde un sistema de archivos local o desde un servidor web.

Clase NetStream

La clase *NetStream* abre una conexión unidireccional de transmisión entre *Flash Player* y *Flash Media Server*, o bien entre *Flash Player* y el sistema de archivos local.

Un objeto *NetStream* es un canal dentro de un objeto *NetConnection*. Este canal puede:

- Publicar una transmisión.
- Suscribirse a una transmisión publicada y recibir datos.
- Publicar o reproducir datos en vivo (en tiempo real) y datos grabados con anterioridad.

También es posible utilizar objetos *NetStream* para enviar mensajes de texto a todos los clientes suscriptores.

La reproducción de archivos de vídeo externos ofrece varias ventajas frente a la incorporación de vídeo en un documento de Flash. Por ejemplo, ofrece un mejor rendimiento y gestión de la memoria, velocidades de fotogramas de vídeo y Flash independientes, etc.

Otro dato importante de esta clase es que proporciona varios métodos y propiedades que permiten controlar el progreso del archivo conforme se carga y reproduce. También proporcionan al usuario control sobre la reproducción (detener, pausa, etc.).

CAPÍTULO 4. DESARROLLO DEL PROYECTO

El objetivo principal de este apartado es mostrar las necesidades del Área de Audiovisuales de la Universidad Carlos III de Madrid con respecto a la nueva aplicación del Gestor de Contenidos Multimedia ARCMM para realizar un archivo de subtítulos para los vídeos existentes en el Gestor.

En esta etapa se comienza por definir los requisitos de usuario, en la fase inicial, así como el diagrama de casos de uso correspondiente.

Después, se explica la fase de análisis, donde se incluyen los requisitos de usuario específicos y su respectivo diagrama de casos de uso. También, se expone el diagrama de clases y la integración de la herramienta de subtitulado con el Gestor ARCMM y la base de datos del Gestor, además de describirse el esquema definido para la creación del archivo de subtítulos.

Posteriormente, se habla de la fase de diseño arquitectónico, describiéndose la infraestructura hardware y software de la aplicación, y la fase de diseño detallado, explicándose la implementación y descripción de las partes del proyecto. Seguidamente, se muestra el producto final puesto en marcha, presentando los resultados del trabajo.

Para terminar, se muestra la planificación de horas dedicadas y un resumen de costes.

4.1. Fase inicial

Como se vio en el capítulo 2.4.3. *Comparación de herramientas*, las herramientas locales poseen muchas de las características importantes para una aplicación de subtítulo, aunque en este caso, estas herramientas no pueden ser utilizadas, ya que habría que instalar la herramienta en los ordenadores de todas las personas que tuviesen el cometido de subtitar los vídeos del portal ARCMM. Además, necesitan que los vídeos se encuentren en el mismo ordenador, aspecto que no puede cumplirse, pues los vídeos se encuentran en servidores.

Por otro lado, la herramienta on-line Overstream podría subtitar los vídeos de ARCMM, ya que sólo precisa la URL del vídeo pero, para ver los vídeos subtitulados, habría que incluir un enlace con la URL del vídeo subtitulado en Overstream. Esto no es deseable ya que, o se muestra el vídeo con los subtítulos visibles todo el tiempo, o habría que presentar dos vídeos, uno con el reproductor de ARCMM sin subtítulos y otro con el de Overstream con los subtítulos. Además, habría que realizar la subtitulación desde fuera del portal del Gestor, aspecto poco útil en este caso, aparte de que Overstream no posee la opción de modificar el color del texto.

Por estas razones, se llevará a cabo el desarrollo de una aplicación propia que permita introducir los datos que forman los subtítulos y visualizarlos para poder modificarlos y ver el posible resultado final con el vídeo, así como un reproductor capaz de “entender” el archivo generado y mostrar los subtítulos. A esta aplicación accederán aquel o aquellos usuarios designados por el Área de Audiovisuales para realizar la edición de subtítulos.

La aplicación y su entorno deben contemplar las siguientes capacidades:

- **Adecuado a estándares:** las páginas desarrolladas deben estar validadas contra el estándar utilizado en el Gestor ARCMM.

- **Facilidad de uso:** la aplicación desarrollada debe ayudar al usuario reduciendo el esfuerzo a la hora de realizar las funciones proporcionadas y presentar una interfaz clara e intuitiva.
- **Integración con el Gestor ARCMM:** la aplicación debe presentar una interfaz compatible con el portal de ARCMM y permitir la comunicación con el Gestor. Además, debe utilizar la infraestructura, herramientas y base de datos que se emplean para el desarrollo de ARCMM, para proporcionar a la aplicación **fiabilidad y escalabilidad**.

4.1.1. Proceso de desarrollo

Para delimitar cada una de las fases de desarrollo de este proyecto, se ha elegido el modelo de Ciclo de Vida en Cascada, abordándose cada una de las fases del proyecto de modo secuencial.

El inicio de cada etapa de este ciclo de vida debe esperar a la finalización de la etapa inmediatamente anterior. Las etapas o fases de este proyecto son:

- **Análisis de requisitos:** se determinan las necesidades de los usuarios finales del software para establecer los objetivos que debe cubrir. De esta fase se obtiene el conjunto formalizado de funcionalidades que debe proporcionar el sistema, sin entrar en detalles internos, y un consenso entre el usuario y el desarrollador de todo lo que se requiere del sistema.
- **Diseño de la arquitectura:** el sistema se descompone y organiza en elementos que puedan elaborarse por separado, incluyendo su distribución en un conjunto de recursos o infraestructura. Tras esta organización, se especifica lo que debe realizar cada una de estas partes y la manera en que se combinan entre ellas.
- **Diseño detallado:** para cada uno de los componentes identificados en la fase anterior, se detalla su funcionalidad, incluyéndose una descripción detallada de sus interfaces y algoritmos.

- **Codificación o Programación:** se implementa el código fuente de cada componente, realizando pruebas para corregir errores que puedan surgir.
- **Pruebas de integración y aceptación:** se ensamblan todos los componentes y se comprueba el correcto funcionamiento de todo el sistema.
- **Implantación y mantenimiento:** el software obtenido se pone en producción. Durante la explotación del sistema, pueden surgir cambios debidos a errores, a que el software deba adaptarse a cambios en el entorno, como el sistema operativo, o a que el cliente requiera ampliaciones funcionales o de rendimiento.

4.1.2. Especificación inicial de requisitos

En este punto se presentan los requisitos que propuso inicialmente el Área de Audiovisuales con respecto a las necesidades que precisaban. En reuniones con el personal del área, se establecieron los siguientes objetivos iniciales que debía cumplir la aplicación:

Especificaciones informales

El objetivo principal de la aplicación es gestionar la subtitulación del contenido audiovisual gestionado por la Universidad Carlos III de Madrid, a través del Área de Audiovisuales. Esto es: su creación, su almacenamiento y su visualización posterior, mediante un reproductor, mientras se reproduce el contenido audiovisual subtitulado.

La aplicación permitirá la introducción de datos por parte del usuario. Los datos a introducir para formar el subtítulo estarán bien especificados y presentarán un formato sencillo, que también deberá estar indicado, para aportar al usuario una mayor facilidad a la hora de generar los subtítulos.

En una región de la interfaz, la aplicación mostrará los subtítulos que se van generando mediante una tabla, cuyas columnas son los datos que introduce el usuario,

para que éste vea los subtítulos de manera clara y esquemática. Además, la aplicación proporcionará la posibilidad de modificar estos subtítulos mediante un icono o botón que indique claramente esta función.

La aplicación también incluirá un reproductor en el que poder visualizar el vídeo del que se crearán los subtítulos. Esto permitirá al usuario la edición de subtítulos de manera sincronizada con el vídeo y, además, ver el posible resultado final de la subtitulación, ya que el reproductor mostrará los subtítulos que se van generando.

Este reproductor presentará una interfaz sencilla con los elementos necesarios para la reproducción de un vídeo. Estos elementos son:

- Botones: botón play, botón pause, botón stop y botón mute.
- Barras: barra de volumen y barra de reproducción, la cual permite posicionarse en un instante del vídeo determinado.
- Etiquetas de tiempo: con el tiempo de reproducción y la duración total del vídeo.

Además, el reproductor incluirá una opción que active y desactive los subtítulos de forma rápida e intuitiva como, por ejemplo, un botón.

Por último, la aplicación presentará un icono o botón que, al ser presionado, guardará toda la información recopilada y formará con ella el archivo de subtítulos que será guardado en el servidor en el que se encuentra el Gestor ARCAMM. El nombre de este archivo será incluido en la base de datos del Gestor.

Además, la aplicación deberá ser integrada en el Gestor ARCAMM y ser, por tanto, compatible con él. Para ello, debe presentar una interfaz acorde con la del Gestor y soportar la lectura de datos que provengan del mismo.

4.1.3. Diagrama de casos de uso inicial

En la *Figura 4.1*, se muestra el diagrama de casos de uso inicial correspondiente con los requisitos iniciales expuestos anteriormente. En esta aplicación sólo existe un rol, el de subtitulador, que será el único que pueda acceder a ella y hacer uso de todas las funciones disponibles.

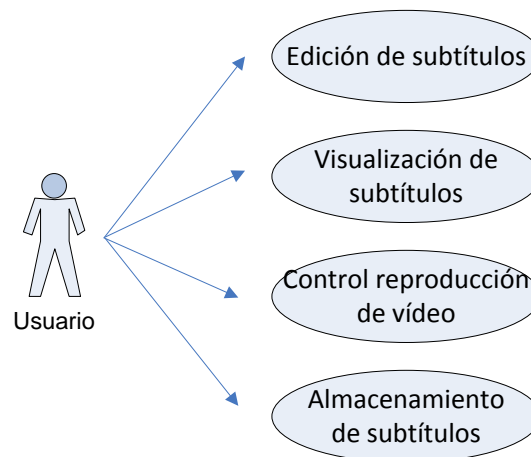


Figura 4.1. Casos de uso inicial

A continuación se ofrece una breve descripción de los casos de uso mostrados en el diagrama de la *Figura 4.1*:

- **Edición de subtítulos:** Permite crear los subtítulos, introduciendo los datos necesarios, y ofrece la posibilidad de modificarlos, ya sean subtítulos en proceso de creación o de un archivo ya existente.
- **Visualización de subtítulos:** Permite visualizar los subtítulos que se crean en el momento o de un archivo de subtítulos existente.
- **Control de reproducción del vídeo:** Se puede controlar la reproducción del vídeo mediante los botones y barras del reproductor, además de mostrar los subtítulos mientras el vídeo se reproduce.
- **Almacenamiento de subtítulos:** Permite guardar los subtítulos creados por el usuario, generando un archivo compatible con el reproductor.

4.2. Fase de análisis

En esta etapa del proyecto se presenta una visión de la fase de análisis, como siguiente paso en el desarrollo de la aplicación. En esta fase se describen los requisitos de usuario y se incluye un diagrama de casos de uso más completo. También se mostrará la integración de la aplicación en el Gestor ARCMM y del archivo de subtítulos en la base de datos del Gestor.

4.2.1. Diagrama de casos de uso en la fase de análisis

En la *Figura 4.2* se representan los Casos de Uso para la fase de análisis.

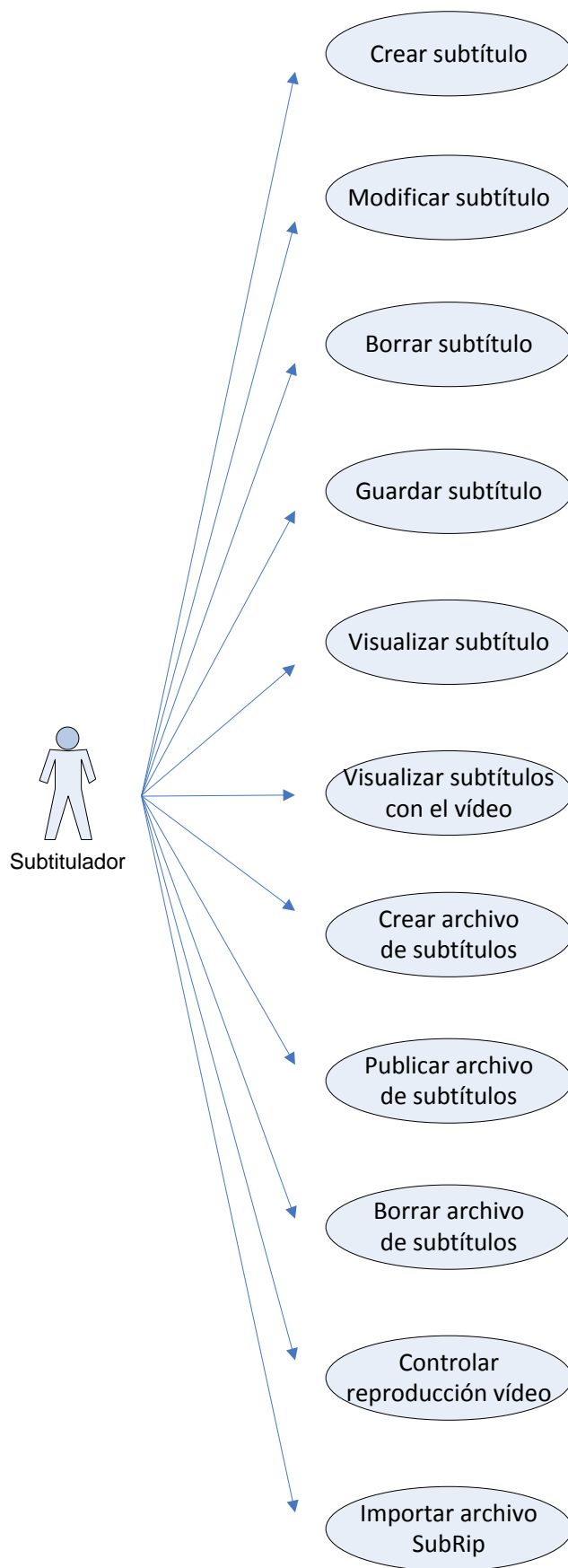


Figura 4.2. Casos de Uso en la fase de análisis

En las siguientes tablas se ofrece una descripción detallada de los casos de uso representados en la *Figura 4.2*. Para cada caso de uso, se incluye una descripción del mismo, las precondiciones (estado del sistema que se tiene que cumplir para que el caso de uso se pueda instanciar), las postcondiciones (estado del sistema una vez instanciado el caso de uso) y el escenario básico (pasos principales del caso de uso ordenados).

Nombre	Crear subtítulo
Descripción	La aplicación debe permitir insertar los datos para un nuevo subtítulo.
Actores	Subtitulador
Precondiciones	Usuario debe validarse en ARCAMM con el rol de Subtitulador.
Postcondiciones	Datos comprobados y guardados.
Escenario	<ol style="list-style-type: none"> 1. Introducir los datos para crear el subtítulo. 2. Pinchar sobre el icono correspondiente para guardar los datos introducidos.

Tabla 4.1. CU-001

Nombre	Visualizar subtítulo
Descripción	La aplicación debe permitir visualizar los subtítulos que se van creando, tras guardarlos.
Actores	Subtitulador
Precondiciones	Debe haberse creado algún subtítulo.
Postcondiciones	
Escenario	<ol style="list-style-type: none"> 1. Mostrar subtítulos.

Tabla 4.2. CU-002

Nombre	Modificar subtítulo
Descripción	La aplicación debe permitir modificar los datos introducidos para cada subtítulo.
Actores	Subtitulador
Precondiciones	Debe haberse creado algún subtítulo.
Postcondiciones	Subtítulo modificado.
Escenario	<ol style="list-style-type: none"> 1. Pinchar sobre el icono “editar” del subtítulo correspondiente. 2. Reescribir los datos. 3. Guardar los nuevos datos. 4. Mostrar los nuevos datos.

Tabla 4.3. CU-003

Nombre	Borrar subtítulo
Descripción	La aplicación debe permitir borrar cualquiera de los subtítulos que se van creando.
Actores	Subtitulador
Precondiciones	Debe haberse creado algún subtítulo.
Postcondiciones	Subtítulo borrado.
Escenario	1. Pinchar sobre el icono “borrar” del subtítulo correspondiente.

Tabla 4.4. CU-004

Nombre	Visualizar subtítulos con el vídeo
Descripción	El reproductor de la aplicación tiene que mostrar los subtítulos que se van creando en los instantes correspondientes, mientras se reproduce el vídeo.
Actores	Subtitulador
Precondiciones	Los subtítulos deben estar activados en el reproductor.
Postcondiciones	
Escenario	1. Leer subtítulos guardados. 2. Mostrar subtítulos durante la reproducción.

Tabla 4.5. CU-005

Nombre	Crear archivo de subtítulos
Descripción	La aplicación tiene que permitir crear un archivo con todos los datos introducidos por el Subtitulador.
Actores	Subtitulador
Precondiciones	Debe haberse creado algún subtítulo.
Postcondiciones	Archivo generado y almacenado en el servidor.
Escenario	1. Pinchar sobre el icono correspondiente para crear el archivo. 2. Generar fichero. 3. Mensaje de confirmación.

Tabla 4.6. CU-006

Nombre	Publicar archivo de subtítulos
Descripción	La aplicación debe permitir hacer público el fichero de subtítulos en ARCMM.
Actores	Subtitulador
Precondiciones	
Postcondiciones	En la página ítem de ARCMM se muestra el vídeo con los subtítulos.
Escenario	1. Seleccionar la opción “Publicar subtítulos”. 2. Guardar la opción activada.

Tabla 4.7. CU-007

Nombre	Borrar archivo de subtítulos
Descripción	La aplicación debe permitir borrar el archivo de subtítulos XML.
Actores	Subtitulador
Precondiciones	Debe existir un archivo de subtítulos.
Postcondiciones	Se elimina el archivo de subtítulos del servidor.
Escenario	1. Pinchar sobre el icono correspondiente para borrar el archivo. 2. Confirmar borrado.

Tabla 4.8. CU-008

Nombre	Controlar reproducción del vídeo
Descripción	El reproductor debe permitir al usuario controlar la reproducción del vídeo: pausarlo/reanudarlo, detenerlo, adelantar/retrasar el instante de reproducción, modificar el volumen o silenciarlo y visualizar el instante de reproducción y la duración total del vídeo.
Actores	Subtitulador
Precondiciones	
Postcondiciones	Reproducción del vídeo modificada según el elemento utilizado.
Escenario	1. Pinchar sobre los botones o barras correspondientes a la acción que se quiere realizar.

Tabla 4.9. CU-009

Nombre	Importar archivo SRT
Descripción	La aplicación debe permitir tomar un archivo externo con formato SubRip y crear, a partir de él, un archivo de subtítulos compatible.
Actores	Subtitulador
Precondiciones	
Postcondiciones	Se genera un archivo de subtítulos que puede ser utilizado por el reproductor.
Escenario	1. Pinchar sobre el icono "Importar" de la aplicación. 2. Seleccionar archivo SubRip del PC. 3. Aceptar la selección.

Tabla 4.10. CU-010

4.2.2. Especificación de requisitos en la Fase de Análisis

En este apartado se muestran los requisitos que debe cumplir la aplicación, generados a partir del diagrama de casos de uso mostrado en el apartado anterior.

RU-001: Formato de aplicación	
Prioridad	Alta
Objetivos	La herramienta debe ser accesible sin necesidad de instalar software adicional. La herramienta debe ser multiplataforma.
Requisitos relacionados	
Descripción	<p>La aplicación debe presentar un interfaz en formato web de manera que sea compatible con los siguientes sistemas operativos:</p> <ul style="list-style-type: none"> - Windows XP, 2000, 2003, Vista, 7 - Linux - Mac OS X <p>La interfaz web debe ser compatible con, al menos, los siguientes navegadores:</p> <ul style="list-style-type: none"> - Internet Explorer 8 - Mozilla Firefox 3 - Chrome 7 - Safari 5

Tabla 4.11. RU-001

RU-002: Rol “subtitulador”	
Prioridad	Alta
Objetivos	Crear nuevo rol “subtitulador” en Gestor ARCAMM
Requisitos relacionados	RU-001: Formato de aplicación
Descripción	Crear un nuevo rol en el sistema ARCAMM de manera que dicho rol sea asignado al módulo de subtitulación.

Tabla 4.12. RU-002

RU-003: Página principal para rol “subtitulador”	
Prioridad	Alta
Objetivos	Crear página principal para el rol “subtitulador”
Requisitos relacionados	RU-002: Rol “subtitulador”
Descripción	Crear una nueva vista en ARCAMM de manera que se tenga acceso directo a los vídeos susceptibles de ser subtitulados.

Tabla 4.13. RU-003

RU-004: Icono identificativo del rol “subtitulador” en página ítem	
Prioridad	Alta
Objetivos	Incorporar icono identificativo del rol “subtitulador” en página ítem
Requisitos relacionados	RU-002: Rol “subtitulador”
Descripción	Proporcionar un acceso directo al módulo de subtitulación una vez que el usuario ha asumido dicho rol.

Tabla 4.14. RU-004

RU-005: Interfaz del módulo de subtitulado	
Prioridad	Media
Objetivos	Interfaz compatible con interfaz de ARCMM
Requisitos relacionados	
Descripción	La interfaz del módulo debe tener un estilo similar a la interfaz de ARCMM, de manera que quede perfectamente integrada en el mismo.

Tabla 4.15. RU-005

RU-006: Creación de formulario para la introducción de datos	
Prioridad	Alta
Objetivos	Permitir al usuario introducir datos para crear los subtítulos
Requisitos relacionados	
Descripción	El usuario debe poder introducir, a través de un formulario, los siguientes datos para crear los subtítulos: <ul style="list-style-type: none"> - Tiempo inicial - Duración - Texto del subtítulo

Tabla 4.16. RU-006

RU-007: Formato de los datos de tiempo	
Prioridad	Alta
Objetivos	Los datos de tiempo deben presentar un formato específico
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	Los datos para generar los subtítulos deben presentar un formato específico. Tiempo inicial en hh:mm:ss.ddd; duración en segundos (pueden representarse con decimales).

Tabla 4.17. RU-007

RU-008: Color de letra	
Prioridad	Media
Objetivos	Permitir diversos colores de letra para cada subtítulo.
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	Permitir al usuario elegir el color de letra para cada subtítulo entre uno de los siguientes: blanco, amarillo, verde, celeste y rosa

Tabla 4.18. RU-008

RU-009: Guardar subtítulos	
Prioridad	Alta
Objetivos	Permitir al usuario guardar el trabajo de subtitulado para poder continuar en un momento posterior con el mismo.
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	Debe existir un icono que permita al usuario almacenar el trabajo realizado y poder continuar posteriormente con el mismo. Los subtítulos se guardarán en un archivo.

Tabla 4.19. RU-009

RU-010: Almacenamiento del archivo de subtítulos	
Prioridad	Alta
Objetivos	Permitir acceso a los subtítulos independientemente de la máquina en la que se trabaje.
Requisitos relacionados	RU-009: Guardar subtítulos
Descripción	El archivo debe almacenarse en un lugar centralizado (servidor de ARCMM) de manera que sea accesible desde cualquier terminal desde la que se continúe el trabajo.

Tabla 4.20. RU-010

RU-011: Disponibilidad de los subtítulos	
Prioridad	Alta
Objetivos	Evitar que los subtítulos estén disponibles antes de la finalización del trabajo.
Requisitos relacionados	
Descripción	Aunque los subtítulos se almacenen, éstos no deben ser visibles para el usuario hasta que se finalice el trabajo de subtitulación.

Tabla 4.21. RU-011

RU-012: Comprobación de la validez de los subtítulos	
Prioridad	Alta
Objetivos	Evitar que se almacenen y se muestren subtítulos no válidos.
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	El sistema debe comprobar que todos los subtítulos están correctamente formateados y son válidos, de manera que no haya campos vacíos o con valores incorrectos.

Tabla 4.22. RU-012

RU-013: Mensajes de error en la validación	
Prioridad	Alta
Objetivos	Avisar al usuario en el caso de que haya algún error durante la validación.
Requisitos relacionados	RU-012: Comprobación de la validez de los subtítulos
Descripción	Se debe mostrar al usuario un mensaje indicando, de manera unívoca, el error encontrado a la hora de validar los subtítulos (tiempos no válidos, texto vacío, etc.).

Tabla 4.23. RU-013

RU-014: Subtítulos superpuestos	
Prioridad	Alta
Objetivos	Evitar subtítulos con tiempos superpuestos.
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	Comprobar a la hora de añadir un nuevo subtítulo que no se superpone con ninguno de los ya existentes.

Tabla 4.24. RU-014

RU-015: Formato de presentación de los subtítulos	
Prioridad	Alta
Objetivos	Mostrar al usuario los subtítulos introducidos de forma clara.
Requisitos relacionados	
Descripción	Se debe mostrar al usuario los subtítulos ya introducidos en forma de tabla, mostrando el texto, el color, el tiempo inicial y la duración del mismo.

Tabla 4.25. RU-015

RU-016: Permitir la modificación de los subtítulos	
Prioridad	Alta
Objetivos	Permitir modificar los subtítulos introducidos.
Requisitos relacionados	RU-009: Guardar subtítulos
Descripción	Debe existir un icono que permita la modificación de los subtítulos introducidos por el usuario. El acceso a esta funcionalidad debe ser independiente para cada subtítulo introducido y debe mostrarse claramente al usuario.

Tabla 4.26. RU-016

RU-017: Cancelar la modificación de los subtítulos	
Prioridad	Media
Objetivos	Permitir cancelar la edición de cada subtítulo.
Requisitos relacionados	RU-016: Permitir la modificación de los subtítulos
Descripción	El usuario debe poder cancelar la edición de un subtítulo y poder volver al modo de añadir nuevos subtítulos sin que se produzca alteración alguna.

Tabla 4.27. RU-017

RU-018: Borrar subtítulos introducidos	
Prioridad	Alta
Objetivos	Permitir eliminar subtítulos previamente introducidos.
Requisitos relacionados	RU-009: Guardar subtítulos
Descripción	Permitir al usuario borrar una entrada completa de un subtítulo de manera que ninguno de los anteriores o posteriores se vean afectados. El acceso a esta funcionalidad debe ser independiente para cada subtítulo y debe mostrarse claramente al usuario.

Tabla 4.28. RU-018

RU-019: Reproductor para visualizar el vídeo a subtitular	
Prioridad	Alta
Objetivos	Reproductor para visualizar el vídeo a subtitular
Requisitos relacionados	
Descripción	La aplicación debe contar con un reproductor de vídeo, de manera que pueda visualizarse previamente los resultados del trabajo de subtitulación.

Tabla 4.29. RU-019

RU-020: Control de la reproducción del vídeo	
Prioridad	Alta
Objetivos	Permitir el control de la reproducción del vídeo mediante la interfaz del reproductor.
Requisitos relacionados	RU-019: Reproductor para visualizar el vídeo a subtítular
Descripción	<p>La interfaz del reproductor debe presentar los siguientes botones y barras que permitan el control de la reproducción:</p> <ul style="list-style-type: none"> - botón play - botón pause - botón stop - botón mute - barra de volumen - barra de reproducción que permita adelantar/retrasar el vídeo - barra de carga del buffer - botón para activar/desactivar los subtítulos <p>También debe mostrar el instante de reproducción y la duración total del vídeo.</p>

Tabla 4.30. RU-020

RU-021: Interfaz del reproductor	
Prioridad	Alta
Objetivos	La interfaz del reproductor debe ser fácil de usar para el usuario
Requisitos relacionados	RU-019: Reproductor para visualizar el vídeo a subtítular RU-020: Control de la reproducción del vídeo
Descripción	La interfaz del reproductor debe diseñarse de manera que cada función proporcionada se muestre de manera clara mediante su botón o barra correspondiente.

Tabla 4.31. RU-021

RU-022: Botón de inserción de nuevo subtítulo	
Prioridad	Alta
Objetivos	Proporcionar al usuario una forma rápida de introducir un nuevo subtítulo.
Requisitos relacionados	RU-006: Creación de formulario para la introducción de datos
Descripción	De cara a agilizar la subtitulación, debe existir un botón de inserción de subtítulo de forma que el sistema rellene automáticamente el campo "Tiempo inicial".

Tabla 4.32. RU-022

RU-023: Finalización del trabajo de subtitulado	
Prioridad	Alta
Objetivos	Permitir al usuario determinar cuando ha terminado el trabajo de subtitulado.
Requisitos relacionados	RU-009: Guardar subtítulo
Descripción	El interfaz debe permitir al usuario determinar cuando ha finalizado el trabajo de subtitulación para hacer visibles los subtítulos a todos los usuarios de ARCMM.

Tabla 4.33. RU-023

RU-024: Archivo de subtítulos XML	
Prioridad	Alta
Objetivos	El archivo de subtítulos debe ser un XML
Requisitos relacionados	RU-009: Guardar subtítulo
Descripción	El archivo de subtítulos debe estar creado con el lenguaje XML y ser compatible con el reproductor usado en ARCMM.

Tabla 4.34. RU-024

RU-025: Esquema de subtitulado	
Prioridad	Alta
Objetivos	Se utilizará un esquema de subtitulado, con un espacio de nombres asignado, para la generación de los subtítulos en XML.
Requisitos relacionados	RU-024: Archivo de subtítulos XML
Descripción	El espacio de nombres representará los datos que definen a cada subtítulo.

Tabla 4.35. RU-025

RU-026: Archivo XML multilenguaje	
Prioridad	Alta
Objetivos	El archivo XML debe ser multilenguaje
Requisitos relacionados	RU-024: Archivo de subtítulos XML
Descripción	La codificación del archivo XML debe permitir el soporte de los caracteres especiales de cualquier idioma oficial en la Unión Europea.

Tabla 4.36. RU-026

RU-027: Borrar el archivo de subtítulos	
Prioridad	Alta
Objetivos	Permitir al usuario borrar la totalidad de subtítulos de un vídeo.
Requisitos relacionados	RU-009: Guardar subtítulos
Descripción	El sistema debe permitir el borrado completo del archivo de subtítulos.

Tabla 4.37. RU-027

RU-028: Solicitar confirmación para borrado	
Prioridad	Media
Objetivos	Evitar el borrado accidental de subtítulos.
Requisitos relacionados	RU-027: Borrar el archivo de subtítulos
Descripción	El sistema debe pedir confirmación a la hora de borrar completamente los subtítulos de un vídeo.

Tabla 4.38. RU-028

RU-029: Reproductor compatible con archivo de subtítulos	
Prioridad	Alta
Objetivos	El reproductor del módulo de subtitulado debe ser compatible con el archivo XML generado.
Requisitos relacionados	RU-024: Archivo de subtítulos XML RU-025: Esquema de subtitulado
Descripción	El archivo XML generado debe poder ser leído por el reproductor del módulo de subtitulado.

Tabla 4.39. RU-029

RU-030: Subtítulos sobre el vídeo	
Prioridad	Alta
Objetivos	El reproductor debe mostrar los subtítulos sobre el vídeo
Requisitos relacionados	
Descripción	El reproductor debe mostrar los subtítulos durante la reproducción del vídeo y de forma que permita la correcta lectura de los subtítulos.

Tabla 4.40. RU-030

RU-031: Subtítulos en tiempo real	
Prioridad	Alta
Objetivos	El reproductor debe mostrar los subtítulos en tiempo real
Requisitos relacionados	RU-030: Subtítulos sobre el vídeo
Descripción	Los subtítulos deben mostrarse de forma inmediata en el reproductor según son introducidos o modificados por el usuario.

Tabla 4.41. RU-031

RU-032: Posicionamiento automático al modificar un subtítulo	
Prioridad	Media
Objetivos	Posicionar la reproducción automáticamente al tiempo inicial del subtítulo a editar.
Requisitos relacionados	RU-016: Permitir la modificación de los subtítulos RU-031: Subtítulos en tiempo real
Descripción	El reproductor de vídeo debe desplazar la reproducción al tiempo inicial de un subtítulo en el momento que se va a modificar el mismo. De esta manera el usuario puede ver el resultado de la modificación en el mismo instante de la modificación.

Tabla 4.42. RU-032

RU-033: Importar archivo externo	
Prioridad	Alta
Objetivos	Permitir importar archivos externos de subtitulación creados con otra herramienta.
Requisitos relacionados	
Descripción	Permitir importar al menos un tipo de formato de subtítulos a la aplicación (formato deseado: srt).

Tabla 4.43. RU-033

RU-034: Comprobar la integridad del archivo a importar	
Prioridad	Alta
Objetivos	Evitar la importación de archivos no soportados.
Requisitos relacionados	RU-033: Importar archivo externo
Descripción	Comprobar antes de realizar la importación que el formato del archivo de externo es compatible con la aplicación.

Tabla 4.44. RU-034

RU-035: Crear interfaz de comunicación entre la aplicación y ARCAMM	
Prioridad	Alta
Objetivos	Permitir que ARCAMM y la aplicación de subtulado intercambien la información necesaria.
Requisitos relacionados	
Descripción	Establecer una interfaz para comunicar el módulo de subtulación con el portal donde se va a integrar.

Tabla 4.45. RU-035

RU-036: Modificar la BBDD de ARCAMM	
Prioridad	Alta
Objetivos	Realizar la ampliación de la BBDD de ARCAMM para permitir el correcto funcionamiento del nuevo módulo.
Requisitos relacionados	
Descripción	Añadir los campos necesarios en la BBDD de ARCAMM de manera que sea compatible tanto con el nuevo módulo, como con los ya existentes.

Tabla 4.46. RU-036

RU-037: Regreso al portal ARCAMM	
Prioridad	Alta
Objetivos	Permitir al usuario subtitulador regresar a la parte de visualización de vídeos de ARCAMM.
Requisitos relacionados	
Descripción	Debe implementarse un icono o botón, claramente visible, que permita al usuario regresar, de forma rápida, a la zona de visualización de vídeos de ARCAMM.

Tabla 4.47. RU-037

RU-038: Cerrar sesión	
Prioridad	Media
Objetivos	Permitir al usuario cerrar su sesión
Requisitos relacionados	
Descripción	Incorporar un icono o botón que permita cerrar la sesión con la que se ha accedido a ARCAMM y situar al usuario en la página principal del gestor.

Tabla 4.48. RU-038

RU-039: Carga automática de subtítulos ya existentes	
Prioridad	Alta
Objetivos	Mostrar automáticamente los subtítulos creados en una sesión anterior.
Requisitos relacionados	RU-009: Guardar subtítulos
Descripción	Si al acceder a la aplicación existe un archivo de subtítulos para el vídeo a subtitular, la aplicación debe mostrarlos, tanto en la tabla como a través del reproductor.

Tabla 4.49. RU-039

RU-040: Nombre del archivo de subtítulos	
Prioridad	Alta
Objetivos	El nombre del archivo XML debe presentar una forma determinada.
Requerimientos	RU-009: Guardar subtítulos
Descripción	El archivo XML debe nombrarse de la siguiente forma: <nombre del vídeo>.xml

Tabla 4.50. RU-040

4.2.3. Diagrama de clases del dominio

En la *Figura 4.3* se muestra el diagrama de clases del dominio obtenido en la fase de análisis del sistema.

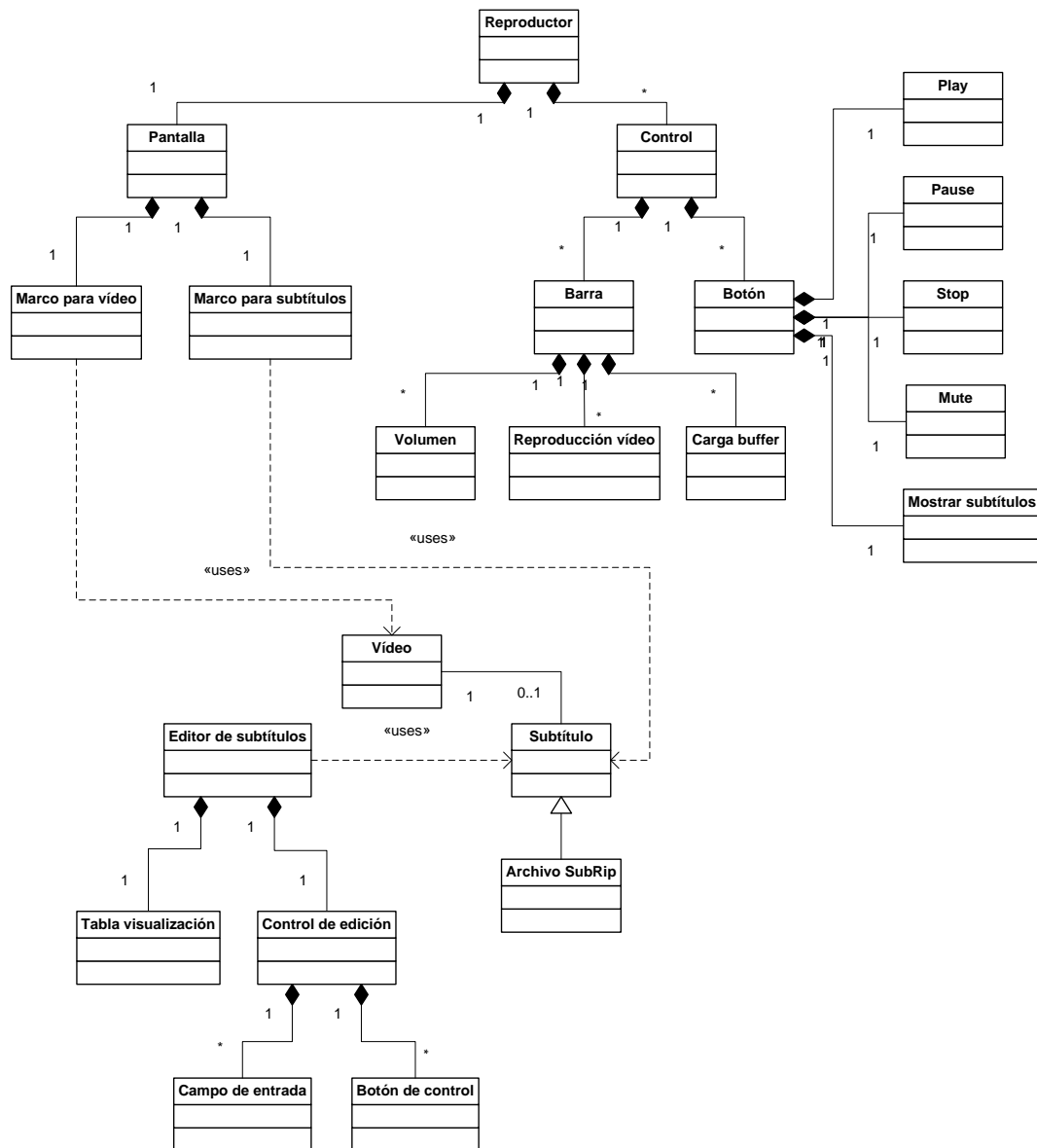


Figura 4.3. Diagrama de clases del dominio

Como puede observarse, el diagrama consta de dos partes: el reproductor y el editor de subtítulos. El reproductor está compuesto de:

- Varios controles: se componen a su vez de barras: la de volumen, la de reproducción del vídeo y la de carga del buffer; y de botones: play, pause, stop, mute y mostrar subtítulos.
- Pantalla: que se compone de un marco para el vídeo, que utiliza el vídeo a subtítular, y un marco para los subtítulos, que utiliza el conjunto de subtítulos para mostrarlo.

Por su parte, el editor de subtítulos está compuesto de una tabla de visualización y un control para la edición. Este control, a su vez, se compone de campos de entrada y botones de control.

El editor de subtítulos genera un conjunto de subtítulos (clase Subtítulo) para el vídeo que muestra el reproductor. Además también se puede utilizar un archivo de subtítulos de formato SubRip. Este archivo se convierte al conjunto de subtítulos que emplea el editor para poder ser utilizado.

4.2.4. Integración en el Gestor ARCMM

Como ya se ha dicho en el apartado 4.1. *Fase inicial*, la aplicación estará integrada en el Gestor ARCMM. Antes de explicar la integración, se hablará brevemente de la manera de acceder al Gestor.

ARCMM permite ver los contenidos multimedia generados por la Universidad Carlos III de Madrid de dos maneras:

- a través de su página web, para cualquier persona.
- accediendo a él mediante un usuario o contraseña, sólo para personal de la Universidad (alumnos y PAS/PDI).

Además, también permite realizar funciones como subir o borrar contenido, gestionar información, etc., aunque esto sólo para el personal de la Universidad seleccionado por los administradores de ARCMM para realizar estas funciones.

Para ello, deben introducir el usuario y contraseña en el Gestor y seleccionar el perfil, denominado rol en el Gestor, específico para las funciones que se desee realizar. Estos roles son asignados por los administradores de ARCMM.

Universidad Carlos III de Madrid

audiovisuales

Hoy es: **Viernes, 02 Julio 2010.**
Mira los actos programados para este mes.

arcomm Media Manager

Inicio Eventos Vídeos/Podcast Series/Cursos RSS ARCA-RSS

Contenidos Multimedia
2506.Horas 5882.

Buscar

Se está migrando el contenido audiovisual desde el antiguo repositorio al nuevo. Si no encuentras el video que buscas, puedes probar a buscarlo [aquí](#)

Usuario:

Contraseña:

Security Code:

Iniciar sesión

RSS Agenda | RSS Video bajo demanda | Este sitio es propiedad de la Universidad Carlos III de Madrid © 2007.
Todos los derechos reservados. Comentarios y sugerencias: audiovisuales@uc3m.es

Figura 4.4. Validación de usuario en ARCAMM

Universidad Carlos III de Madrid

audiovisuales

Hoy es: **Lunes, 05 Julio 2010.**
Mira los actos programados para este mes.

arcomm Media Manager

Inicio Eventos Vídeos/Podcast Series/Cursos RSS ARCA-RSS SONIA MARTIN CARMONA

Contenidos Multimedia
2507.Horas 5882.

Buscar

Se está migrando el contenido audiovisual desde el antiguo repositorio al nuevo. Si no encuentras el video que buscas, puedes probar a buscarlo [aquí](#)

Seleccione un rol:

Iniciar sesión

RSS Agenda | RSS Video bajo demanda | Este sitio es propiedad de la Universidad Carlos III de Madrid © 2007.
Todos los derechos reservados. Comentarios y sugerencias: audiovisuales@uc3m.es

Figura 4.5. Elección de roles en ARCAMM

Para registrarse en ARCAMM se realiza una validación contra LDAP (*Lightweight Directory Access Protocol*) [OpenLDAP, 2010] y, tras elegir el rol, se pasa a la página

principal del rol elegido (si sólo se posee un rol, se muestra directamente su página principal, sin pasar por la elección de roles). Al seleccionar un contenido multimedia, se pasa a la página *ítem*, en la que se muestran los datos de ese elemento multimedia, que permite visualizarlo y, en su caso, descargarlo. Además, en esta página aparecen las funciones que pueden realizarse con el rol escogido. En la *Figura 4.6*, se puede ver un ejemplo de la página ítem para el rol Manager.

Figura 4.6. Página ítem de ARCAMM para el rol Manager

Para la aplicación desarrollada en este proyecto se ha creado un nuevo rol *subtitulador* y su página correspondiente. Así, tras validarse y elegir el rol subtitulador, se pasará a una página en la que aparecerán los vídeos de la página principal de ARCAMM, susceptibles de ser subtitulados.

Al seleccionar un vídeo, se pasará a la página ítem, en la que aparecerá un icono que representa la función de subtítular. Al pinchar en el icono se accederá a la aplicación de subtitulado, pasándose unos valores por la URL: el nombre del vídeo y el nombre de un archivo de subtítulos temporal donde se guardarán los subtítulos que se van generando hasta crear el archivo final. El nombre del archivo temporal será el identificador del vídeo a subtítular, para que sea único para cada vídeo.

En la página ítem se realiza una consulta a la base de datos de ARCMM para tomar algunas de las características o atributos del vídeo con el objeto de ser mostrados en la página. De esta consulta se obtendrá el nombre e identificador del vídeo para pasarlos a la aplicación de subtitulado.

4.2.5. Comunicación con la base de datos de ARCMM

Para esta aplicación se utilizará la base de datos existente para el Gestor ARCMM, ya que el nombre del archivo de subtítulos que se genera debe ser incluido en ella como otro dato de los elementos multimedia.

La base de datos está formada por varias tablas. En este caso se ha utilizado la *tabla ítem*, cuyas *tuplas* identifican a cada uno de los elementos multimedia almacenados. Estas tuplas incluyen valores en varias columnas o campos que se corresponden con atributos y características del elemento multimedia, como el nombre, fecha en la que se subió al servidor, autor, canal al que pertenece, etc. A continuación se muestra la tabla Ítem aunque, por motivos de seguridad, sólo se muestran algunos de sus campos.

TABLA ÍTEM	ID	ID_channel	title	visited	description	author	subs	...
------------	----	------------	-------	---------	-------------	--------	------	-----

Tabla 4.51. Tabla Ítem de la base de datos

Para esta aplicación, la tabla Ítem ha sido modificada, creándose una columna nueva para guardar el nombre del archivo de subtítulos: *subs_url*. Este campo se

consultará para saber si existe el archivo y poder mostrarlo a la vez que el elemento multimedia, siempre y cuando se cumpla otra condición, que se explicará más adelante. La consulta se realizará en la página ítem (explicada en el apartado 4.2.4. *Integración en el Gestor ARCAMM*).

Si existe un archivo de subtítulos y se quiere acceder al módulo de subtitulado, el nombre de este archivo, obtenido de la consulta a la base de datos, se pasará a dicho módulo a través de la URL (junto con el nombre del vídeo y el archivo temporal). Al acceder al módulo, el archivo es visualizado, tanto en la tabla de visualización como con el vídeo, y puede ser modificado si se desea.

Además, entre las columnas de cada elemento multimedia, existe una útil para este proyecto: *subs*. Esta columna indica si el elemento multimedia posee subtítulos (antes de este proyecto, los subtítulos eran imágenes superpuestas al vídeo o presentación).

El valor de *subs* es lo que se utilizará para indicar si el archivo de subtítulos es público o no. Así, si el valor de *subs* es 0, el archivo de subtítulos no es público, y si el valor es 1, sí lo es.

Con esta información ya se puede saber si el archivo de subtítulos puede mostrarse durante la reproducción del contenido multimedia en la página ítem, es decir, si el archivo existe y es público, se mostrarán los subtítulos.

Si se genera un fichero de subtítulos nuevo o se realizan modificaciones sobre uno existente, al terminar de crear el archivo se accederá a la base de datos para modificar el campo *subs_url* introduciendo el nombre. Si además de crear el archivo, el usuario selecciona la opción “Publicar subtítulos”, significará que los subtítulos deben hacerse públicos y, por tanto, también se cambiará el campo *subs* al valor 1.

Por último, si el usuario decide borrar un archivo de subtítulos existente, se accederá a la base de datos para borrar el nombre del archivo del campo *subs_url* y se modificará el campo *subs* al valor 0.

4.2.6. Archivo de subtítulos

Los subtítulos se guardarán en un archivo que será almacenado en el servidor donde se encuentra la aplicación ARCAMM. Este archivo contiene los datos que caracterizan a cada subtítulo y será utilizado por el reproductor que se creará en este proyecto.

Para poder almacenar los subtítulos se ha elaborado un esquema de metadatos que permite generar archivos XML según el formato que define.

A continuación se muestra el diagrama que representa este esquema, denominado Esquema de Subtítulos de la Universidad Carlos III de Madrid, ESUC3M:

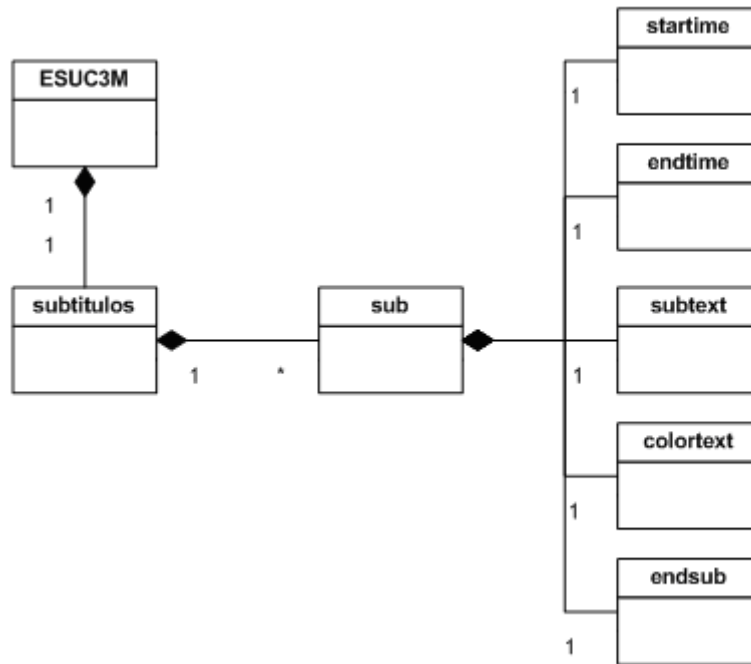


Figura 4.7. Esquema del archivo de subtítulos

Como se muestra en la *Figura 4.7*, el archivo de subtítulos está compuesto de un elemento que comprende todos los subtítulos generados por el usuario. A su vez, cada uno de los subtítulos está compuesto por los siguientes atributos: tiempo de entrada (*starttime*), tiempo de salida (*endtime*), texto (*subtext*), color de letra (*colortext*) y, además, un elemento (*endsub*) que indica si es o no el último subtítulo del archivo.

Otro dato importante es que este archivo debe generarse de tal forma que permita la codificación de múltiples idiomas, es decir, que sea compatible con todo tipo de caracteres, por este motivo se ha utilizado una codificación UTF-8.

Se ha generado un vocabulario de metadatos específico para representar subtítulos debido a que cuando se identificó que era necesario, no existía ningún vocabulario estandarizado y los existentes eran, además, demasiado complejos para las necesidades de la aplicación.

No obstante, con el fin de proporcionar compatibilidad con los vocabularios de subtítulos existentes o que puedan surgir y con formatos existentes, se ha previsto el desarrollo de analizadores sintácticos que, añadidos a la aplicación, permitirían la importación/exportación de subtítulos siguiendo dichos vocabularios o formatos. A modo de ejemplo, la aplicación ha incluido el analizador correspondiente al formato SubRip.

4.3. Fase de diseño arquitectónico

Esta etapa del proyecto comprende, en primer lugar, la última versión de la arquitectura definida para el soporte de la aplicación y la descripción de su diseño e implementación.

En esta sección se proporciona, primeramente, una visión global de la descomposición en componentes del sistema así como del despliegue de los mismos. Seguidamente se describirá de forma detallada la funcionalidad de la aplicación, tomando como referencia la descripción del interfaz desarrollado.

En este proyecto se ha utilizado un estilo arquitectónico para el desarrollo de aplicaciones web en capas. En este caso, los componentes desarrollados se han dividido en dos capas: Presentación, para la parte de la interfaz y el control de la aplicación, y Modelo, para la parte de la funcionalidad.

PRESENTACIÓN	JavaScript, PHP5
MODELO	PHP5, Adobe Flash, ActionScript 3.0
INFRAESTRUCTURA	Librerías de ARCMM
PERSISTENCIA	MySQL

Figura 4.8. Capas de la arquitectura

Los lenguajes de programación a utilizar fueron especificados por el personal del Área de Audiovisuales con el fin de facilitar el proceso de integración con el Gestor ARCMM, esto es, lenguajes compatibles con los que se usaron para desarrollar dicho Gestor. Concretamente: para la capa Presentación, se han utilizado los lenguajes PHP5 y JavaScript; para la capa Modelo, se ha empleado PHP5 en la implementación del editor de subtítulos y Adobe Flash y el lenguaje ActionScript, en su versión 3.0, en la implementación del reproductor.

Para la creación del archivo de subtítulos generado con el editor de subtítulos, se ha utilizado el lenguaje de etiquetado XML. Para este archivo se ha generado un nuevo espacio de nombres siguiendo el esquema ESUC3M, definido en el apartado 4.2.5. *Archivo de subtítulos*, mediante un DTD que define los elementos que representan a cada subtítulo en un formato compatible con el reproductor.

Finalmente, para la comunicación entre el editor y la base de datos de ARCMM, se ha empleado el lenguaje SQL [Gennik, 2004].

4.3.1. Infraestructura Software

En la *Figura 4.9*, se muestra el esquema de la arquitectura software de la aplicación. Como se puede observar, el sistema se divide en las capas Vista y Modelo.

En la capa Presentación está la Interfaz de Usuario de la aplicación, generada dinámicamente mediante PHP y JavaScript, lenguajes que se comunican entre sí para mostrar la página en el navegador web.

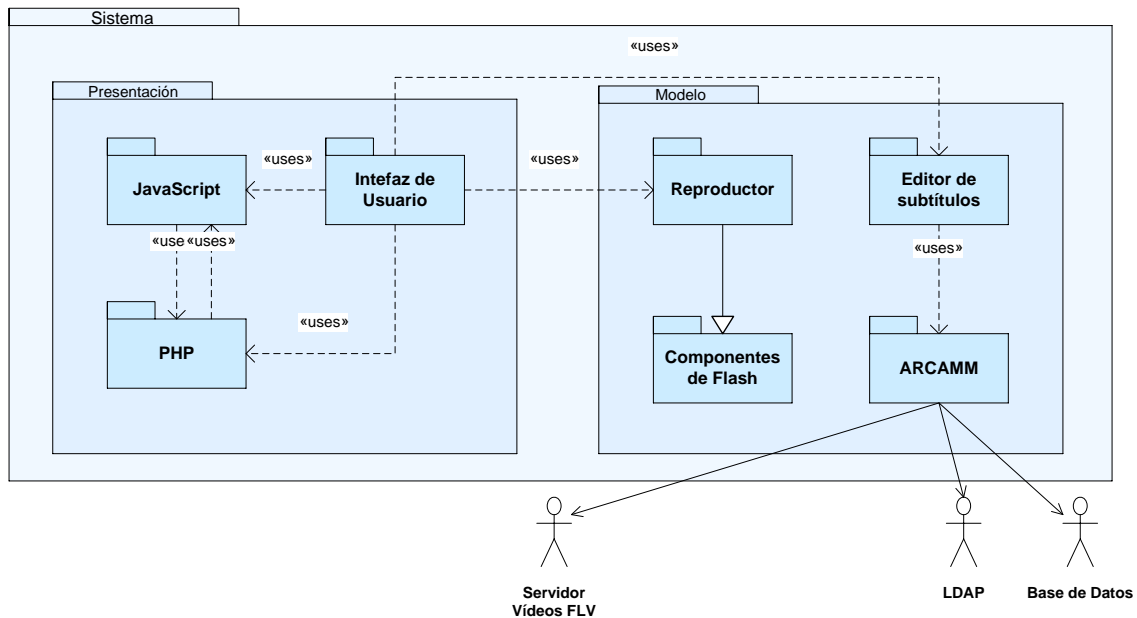


Figura 4.9. Arquitectura Software

En la capa Modelo se encuentran los paquetes que representan a cada uno de los elementos funcionales del sistema: reproductor, editor y componentes de ARCAMM. El reproductor está formado por varios elementos que son una generalización de ciertos componentes de Adobe Flash.

El reproductor y el editor se comunican a través de la Interfaz de Usuario, pues es ésta la que posee los elementos de control que relacionan a ambos.

Además, el editor utiliza componentes de ARCAMM: la validación contra LDAP, la selección de roles, la vista de los objetos multimedia y la página ítem de cada objeto multimedia.

A través de ARCAMM, el editor también accede a la base de datos para tomar algunas características del objeto audiovisual a subtitular y para introducir el nombre del archivo de subtítulos y modificar el campo de subtítulos existente.

Por su parte, el reproductor toma de ARCAMM el vídeo que se va a utilizar, el cual se obtiene del Servidor de Vídeos FLV (servidor en el que se encuentran almacenados los vídeos en formato .flv).

4.3.2. Infraestructura Hardware

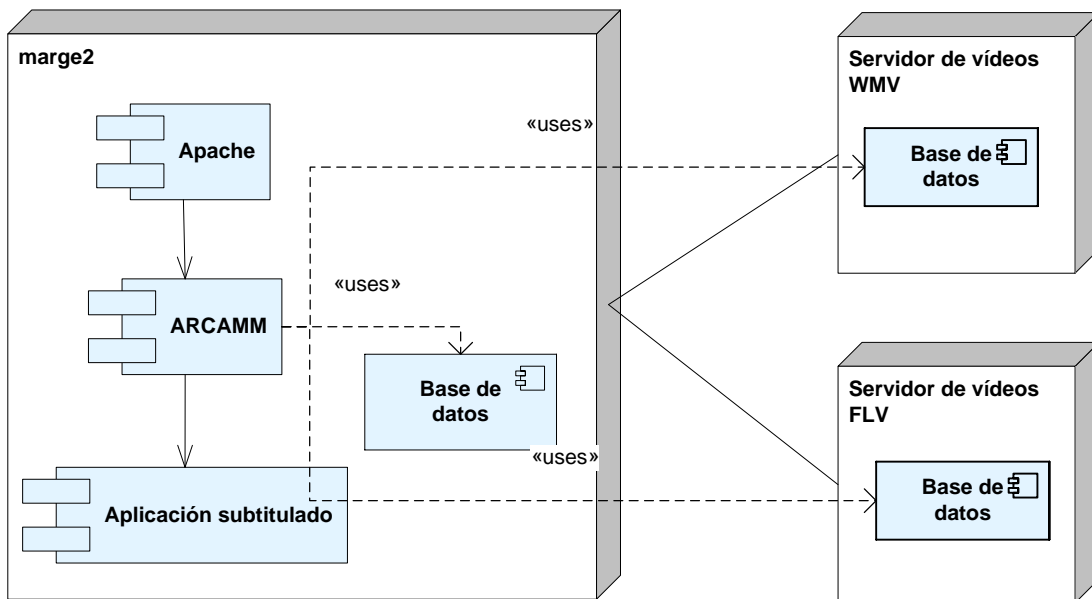


Figura 4.10. Arquitectura Hardware

El editor de subtítulos se encuentra en la misma máquina en la que se aloja el Gestor ARCAMM para obtener un mejor aprovechamiento de recursos y mayor seguridad. Así, la arquitectura de la aplicación será la misma que presenta ARCAMM.

Como se puede observar en la *Figura 4.10*, en la máquina *marge2* de la Universidad Carlos III de Madrid, en la que se tiene como servidor web a Apache, se encuentra el Gestor ARCAMM y la base de datos que éste utiliza. Esta máquina se conecta a otros dos servidores para obtener los vídeos que se muestran en ARCAMM, uno para los vídeos en formato de Flash (extensión .flv) y otro para los vídeos en formato de Windows Media (extensión .wmv).

La aplicación de subtitulado se conecta con ARCAMM al ser una herramienta integrada en el Gestor, y emplea, a través de ARCAMM, la base de datos para tomar valores de ella o hacer modificaciones en algunos de los campos. Además, también se conecta, al igual que antes, a través de ARCAMM, con el servidor de vídeos FLV del que obtiene los vídeos que se muestran en el reproductor. Sólo se comunica con el servidor de vídeos FLV porque éstos son los vídeos susceptibles de subtitularse.

4.3.3. Reproductor

En la *Figura 4.11*, se muestran los componentes utilizados para crear el reproductor y los componentes de Flash a los que especializan.

Para desarrollar los componentes del reproductor para la aplicación de subtitulado se han utilizado componentes existentes en Flash y se les ha añadido la funcionalidad necesaria para esta aplicación:

- **Pantalla de vídeo:** se utiliza el componente Vídeo de Flash y se le añade la función de cargar el vídeo indicado por el nombre que se toma de la URL del editor.
- **Botón “Añadir subtítulos”:** se utiliza el componente *Button* de Flash, añadiéndole la función de pausar el vídeo y tomar el instante de reproducción del mismo.
- **Botón Play/Pause:** se toma la propiedad *PlayPauseButton* del componente *FLVPlayback* de Flash y se incorpora la función de reproducir/pausar el vídeo, así como los cambios de apariencia entre los distintos estados de un botón (botón sin utilizar, ratón encima del botón y botón pulsado).
- **Botón Stop:** similar al botón anterior, con la diferencia de que se toma la propiedad *StopButton* y la funcionalidad añadida es la de detener el vídeo y situar la reproducción en el principio.
- **Botón Mute:** como los botones anteriores, con la excepción de que se toma la propiedad de *MuteButton* y se añade la función de silenciar el sonido.

- **Botón Caption:** en este caso se toma la propiedad *CaptionButton* del componente *FLVPlayback Captioning* y se añade la funcionalidad de mostrar los subtítulos en la pantalla de vídeo.
- **Barra Volumen:** se toma la apariencia de la propiedad *VolumeBar* de *FLVPlayback* y se le da la función de aumentar/disminuir el volumen del sonido.
- **Barra Reproducción/Carga buffer:** se toma la apariencia de la propiedad *SeekBar* para mostrar el proceso de reproducción y se añade la misma apariencia para la carga del buffer, proporcionándole, además de la funcionalidad de cambiar a medida que se reproduce el vídeo, la función de avanzar/retroceder el tiempo del vídeo.

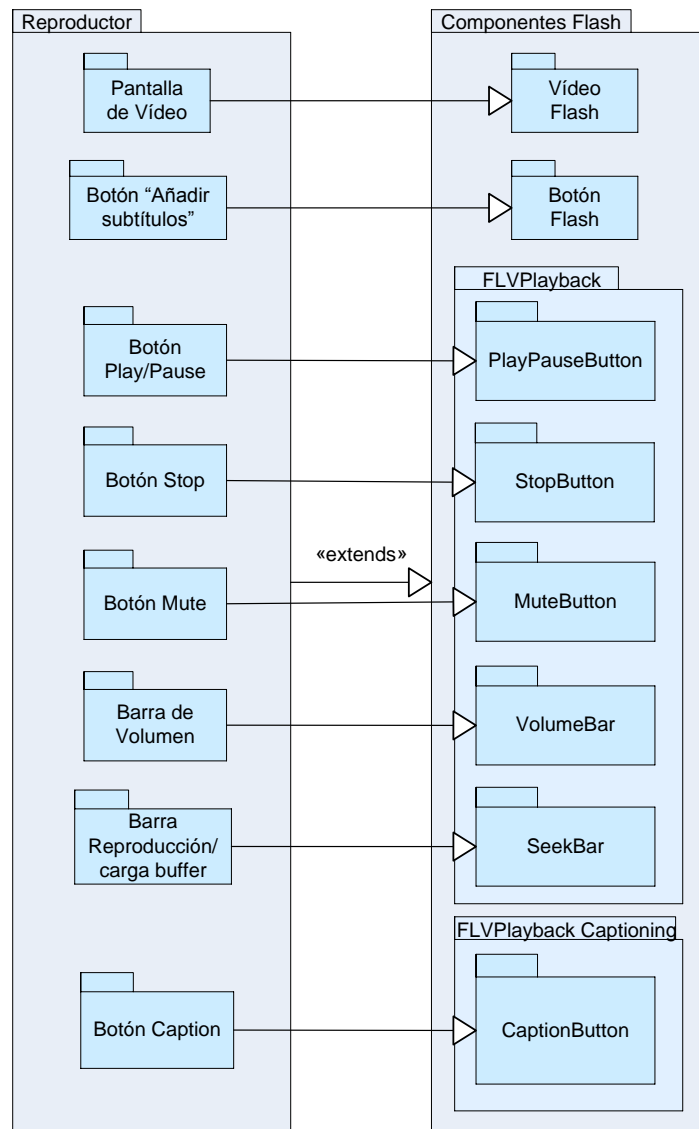


Figura 4.11. Reproductor-Componentes de Flash

4.3.4. Editor de subtítulos

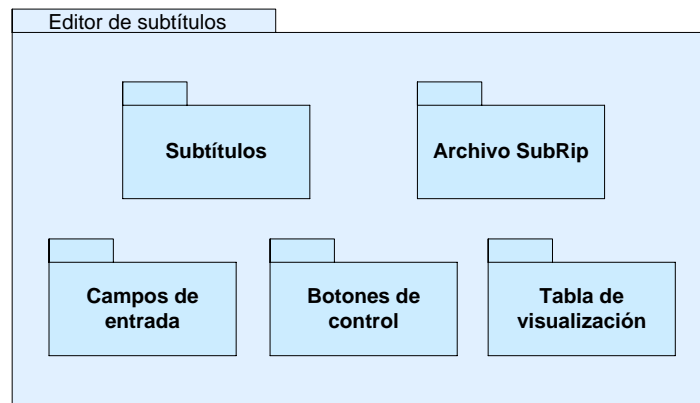


Figura 4.12. Arquitectura del Editor de subtítulos

Los componentes utilizados en la parte del editor son:

- **Subtítulos:** son los datos que introduce el usuario para formar cada subtítulo. Estos subtítulos se guardan en un archivo y pueden ser visualizados, editados y/o borrados. El archivo generado es leído por el reproductor para mostrar los subtítulos mientras se reproduce el vídeo. Si al acceder a la aplicación, ya existe este archivo, se mostrará toda la información para que pueda ser modificada si se desea.
- **Archivo de subtítulos SubRip:** es un archivo de subtítulos con extensión .srt, creado con otra herramienta de subtitulado, que puede importarse para su visualización y edición con el editor de subtítulos desarrollado.
- **Campos de entrada:** recogen los datos que introduce el usuario para formar los subtítulos.
- **Botones de control:** botones o iconos que emplea el usuario para interactuar con el editor. Realizan acciones como guardar, editar y borrar los subtítulos, crear el archivo de subtítulos, importar el archivo SubRip, etc.
- **Tabla de visualización:** muestra los subtítulos que el usuario genera, indicando los datos que caracterizan a cada subtítulo. Si al acceder a la aplicación, ya existe un archivo de subtítulos, muestra el contenido de dicho archivo para que pueda ser modificado.

4.3.5. Archivo XML de subtítulos

Para crear el archivo de subtítulos, del que se habló en el apartado 4.2.6. *Archivo de subtítulos*, se ha utilizado el lenguaje XML, ya que es un lenguaje sencillo de utilizar y fácil su extensión si es necesario incluir nuevas características.

Para este proyecto se ha creado un nuevo vocabulario de metadatos a partir del cual pueden generarse los ficheros de subtítulos en <http://marge2.uc3m.es/arcamm/uri.html>.

La DTD del archivo se muestra a continuación:

```
<!ELEMENT subtítulos (sub+)>
<!ELEMENT sub (starttime, endtime, subtext, colortext, endsub)>
<!ELEMENT starttime (#PCDATA)>
<!ELEMENT endtime (#PCDATA)>
<!ELEMENT subtext (#PCDATA)>
<!ELEMENT colortext (#PCDATA)>
<!ELEMENT endsub (#PCDATA)>
```

Ejemplo 4.1. DTD del archivo de subtítulos

El elemento “subtítulos” representa al conjunto de subtítulos y comprende a varios elementos “sub”, que representan a cada uno de los subtítulos por separado. Este elemento debe aparecer al menos una vez. A su vez, “sub” comprende los elementos:

- **starttime**: tiempo de entrada del subtítulo representado en segundos (puede incluir decimales).
- **endtime**: tiempo de salida del subtítulo representado en segundos (puede incluir decimales).
- **subtext**: texto del subtítulo.
- **colortext**: color de letra del texto, representado por la inicial de cada color: B=Blanco, Am=Amarillo, V=Verde, Az=Azul y R=Rosa.

- **endsub**: valor que indica si es el último subtítulo: el valor 0 indica que no es el último y el valor 1 indica que sí lo es.

Estos elementos deben aparecer sólo una vez para cada elemento “sub”.

Para permitir cualquier carácter en el texto del subtítulo y facilitar la compatibilidad con cualquier idioma, se usa una codificación UTF-8 en el archivo.

A continuación se muestra un ejemplo de este archivo:

```
<?xml version="1.0" encoding="UTF-8">
<subtitulos>
  <sub>
    <starttime>4.208</starttime>
    <endtime>7.895</endtime>
    <subtext>Aquí entradas para el Expreso de Chatsworth.</subtext>
    <colortext>B</colortext>
    <endsub>0</endsub>
  </sub>
  ...
  <sub>
    <starttime>27.774</starttime>
    <endtime>31.122</endtime>
    <subtext>Vengan a ver sin molestia...</subtext>
    <colortext>Am</colortext>
    <endsub>1</endsub>
  </sub>
</subtitulos>
</xml>
```

Ejemplo 4.2. Ejemplo de código del archivo de subtítulos

4.4. Fase de diseño detallado

En este apartado se describirá cómo se ha implementado la funcionalidad requerida para el reproductor y para el editor de subtítulos, así como el proceso de integración con el Gestor ARCAMM.

Hay que tener en cuenta que la aplicación de subtitulado desarrollada en este proyecto incluye el reproductor, es decir, al hablar de la descripción de la aplicación, se explicará el funcionamiento de la edición de subtítulos con el reproductor incluido, pero para entender mejor el funcionamiento del reproductor, se contará su implementación por separado.

4.4.1. Implementación del reproductor

En la *Figura 4.13* se puede ver los pasos para crear el reproductor y ponerlo en marcha de forma esquemática.

Tras realizarse la inicialización del reproductor, se comprueba si se ha realizado la conexión con el servidor de vídeos. Si la conexión es satisfactoria, ya se puede tomar el vídeo y activar todos los botones y barras de la interfaz del reproductor para realizar el control de la reproducción. Posteriormente se comprueba si existe el archivo XML. En caso afirmativo, además de reproducir el vídeo, se muestran los subtítulos.

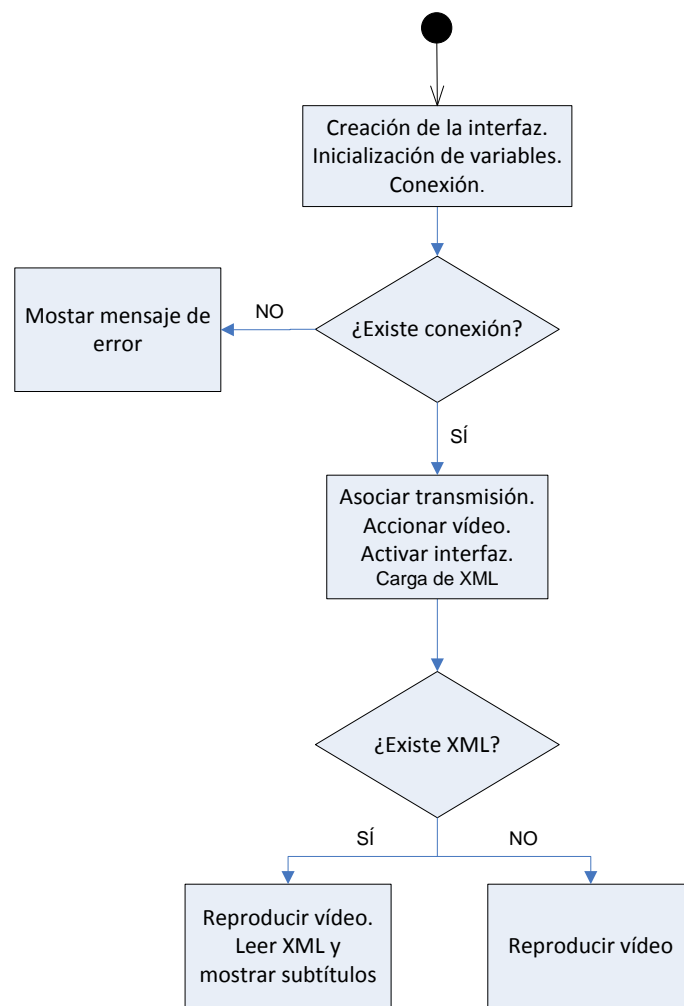


Figura 4.13. Flujo del funcionamiento del reproductor

A continuación se explica detalladamente el desarrollo del reproductor. Para ello, lo dividiremos en cuatro módulos: inicialización, pantalla de vídeo, interfaz y lectura/escritura de subtítulos.

4.4.1.1. Inicialización

En este primer módulo, primero se crea la pantalla para el vídeo y la interfaz de control (elementos y botones útiles para la reproducción) y se realiza la inicialización de variables necesarias para el subtitulado y reproducción. Posteriormente se toma del Gestor ARCAMM el nombre del vídeo y del archivo XML (en el caso de que exista) y se comprueba si existe conexión con el servidor del que se tomará el vídeo a reproducir.

Si la conexión al servidor es satisfactoria, se asocia una transmisión a esa conexión, por la que se enviará el vídeo, y se añade a la pantalla, donde se visualizará el vídeo. También se acciona la reproducción del archivo de vídeo (aunque se detiene para que el usuario lo inicie cuando desee) y se activan los elementos de la interfaz para que puedan utilizarse.

4.4.1.2. Pantalla de vídeo

Muestra el vídeo seleccionado. Si ha sido subtitulado, en la parte inferior de la imagen se mostrarán los subtítulos del archivo XML. Esta funcionalidad se activará mediante el botón *Caption* de la interfaz.

La pantalla de vídeo también incluye un componente que permite iniciar la reproducción al pinchar sobre ella.

4.4.1.3. Interfaz de control

Se compone de los botones y otros elementos que se encargan de controlar la reproducción del vídeo y de mostrar los subtítulos. Los elementos utilizados son:

- **Botón Play/Pause:** reproduce/pausa el vídeo.
- **Botón Stop:** detiene el vídeo y sitúa la reproducción al principio de la misma.
- **Botón Mute:** activa/desactiva el silenciado (apagado total del sonido).
- **Barra de Volumen:** aumenta/disminuye el volumen del sonido a gusto del usuario.
- **Barra de reproducción del vídeo y carga del buffer:** muestra, de forma visual, en qué instante está la reproducción del vídeo y la carga del buffer de reproducción. Además, permite al usuario adelantar/retrasar la reproducción del vídeo.
- **Botón Caption:** activa/desactiva la visión de subtítulos.
- **Etiqueta de tiempos:** muestra el instante de reproducción y la duración del vídeo.
- **Botón “Añadir subtítulos”:** detiene el vídeo y toma el instante de reproducción en el que se detuvo para que pueda ser utilizado en la edición de subtítulos.

Además de las funciones principales, estos elementos realizan otras funciones que se describirán a continuación.

4.4.1.3.1. Botones de reproducción

Son los botones Play/Pause, Stop, Caption y Mute. Estos botones presentan tres estados:

- Pulsado¹⁰: presenta un aspecto como el de la *Figura 4.14 (1)*.
- Encima (*over*): presenta un aspecto como el de la *Figura 4.14 (2)*.
- En reposo: presenta el aspecto de la *Figura 4.14 (3)*.



Figura 4.14. Ejemplo de estados para el botón Pause

La funcionalidad de los botones se realiza en el estado “Pulsado”, pero los otros estados son importantes para una correcta interacción con el usuario. A continuación se detallan las funciones de cada estado:

- **En reposo**: se ejecuta al mostrarse el botón por primera vez y cuando el usuario no interacciona con él. Su función es indicar que el botón no se está utilizando.
- **Encima**: se ejecuta cuando el usuario coloca el ratón sobre el botón. Al cambiar de color, el usuario sabe que está situado encima del botón correspondiente y que éste funciona correctamente.
- **Pulsado**: se ejecuta al hacer *clic* en el botón. Como se comentaba anteriormente, en este estado se realizan las funciones específicas asociadas a los botones.

A continuación se explicará cada botón por separado.

¹⁰ Hay que aclarar que este estado presenta dos posiciones: presionar y levantar. Al presionar, se muestra el Botón Pulsado y al levantar, se realizan las demás acciones, pues se entiende que éstas se ejecutan una vez se ha terminado de presionar el botón. Para simplificar su descripción, se hará referencia a este estado como el conjunto de ambas posiciones.

Botón Play/Pause

Este botón contiene a los botones Play y Pause, ya que cuando se muestra uno de ellos, el otro queda oculto. Cada uno de los botones presenta unas funciones diferentes.

Al crear el reproductor, el vídeo queda detenido para que el usuario lo inicie cuando desee, por lo que el botón que aparece en el interfaz es el Play. Para iniciar la reproducción, el usuario coloca el ratón sobre el botón, mostrándose la imagen del estado Encima. Al hacer clic, se muestra el estado Pulsado y comienza la reproducción del vídeo, con lo que también es necesario dejar de mostrar el elemento play sobre el vídeo y mostrar el botón Pause para poder detener la reproducción.

Por su parte, el botón Pause, tiene el mismo funcionamiento, sólo que al ser pulsado, detiene la reproducción. A la vez, muestra de nuevo el botón Play que continua la reproducción en el instante en el que ésta se había detenido.

En la *Figura 4.15*, se muestra un diagrama de estados de este botón.

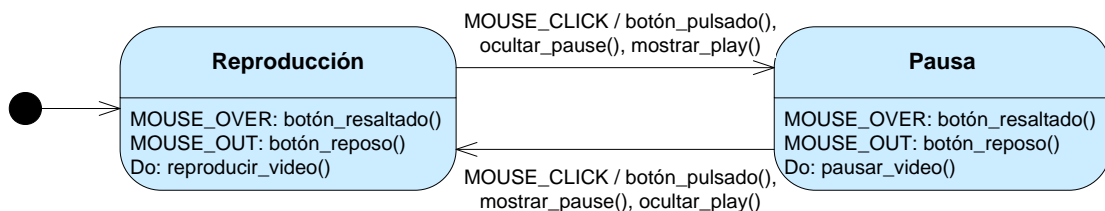


Figura 4.15. Diagrama de estados del botón Play/Pause

Botón Stop

Al pinchar sobre este botón, se detiene la reproducción, situándola al comienzo de la misma. Además, muestra el botón Play y habilita la funcionalidad de inicio de reproducción sobre la pantalla de vídeo, para que el usuario reinicie la reproducción.

Además de la funcionalidad anterior, también deja de mostrar los subtítulos (si se estaban mostrando) y sitúa la lectura del archivo XML al principio, para poder comenzar la reproducción y la lectura a la vez.

En la *Figura 4.16*, se muestra un diagrama de estados del botón.

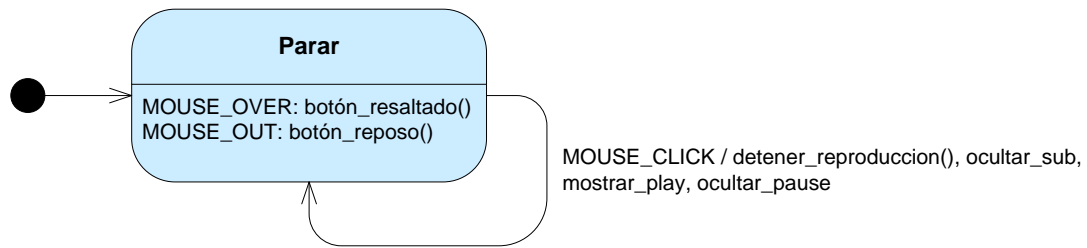


Figura 4.16. Diagrama de estados del botón Stop

Botón Mute

Está compuesto por dos botones, el que indica que el sonido está silenciado, al que se le llamará Mute Off, y el que indica que el sonido está activo, Mute On.

Al crearse la interfaz, el botón que aparece es Mute On, iniciándose la reproducción con el sonido activado. Si el usuario se sitúa sobre él, aparte de mostrarse la imagen del estado Encima, aparece la barra de Volumen. Si se pulsa, aparece Mute On Pulsado y se desactiva (silencia) el sonido. Seguidamente, se muestra Mute Off y deja de mostrar la barra de volumen, al no poder modificarse el volumen cuando el sonido está desactivado.

Para volver a activar el sonido, es preciso pulsar Mute Off, lo que vuelve a mostrar Mute On.

Se puede ver el diagrama de estados en la *Figura 4.17*.

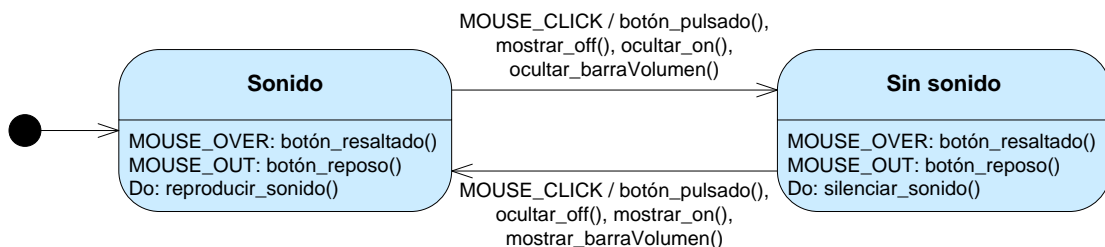


Figura 4.17. Diagrama de estados del Botón Mute

Botón Caption

Está compuesto por los botones Caption On, que indica que se están mostrando los subtítulos, y Caption Off, para saber que no se están mostrando.

El botón que aparece al crearse el interfaz es Caption Off, pues los subtítulos están desactivados por defecto. Al pulsar en Caption Off, aparecerán los subtítulos y el botón Caption On, indicando que están activos.

Al pulsar en Caption On, desaparecerán los subtítulos y se mostrará de nuevo el botón Caption Off.

En la *Figura 4.18* se muestra el diagrama de estados del botón.

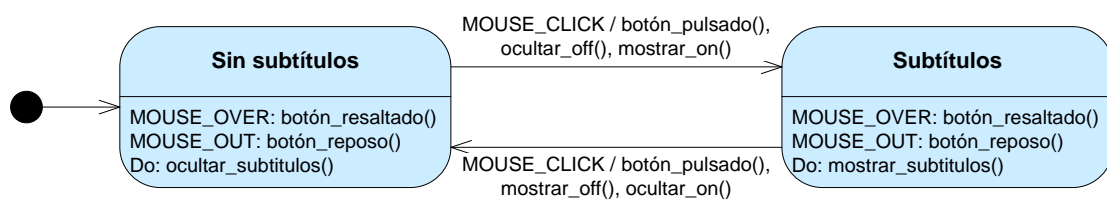


Figura 4.18. Diagrama de estados del Botón Caption

4.4.1.3.2. Barra de Volumen

Esta barra se muestra cuando el usuario sitúa el ratón sobre el Botón Mute On. Al situarse sobre ella, permite modificar el volumen del sonido del vídeo deslizando la zona oscura.



Figura 4.19. Barra de Volumen

La barra puede desaparecer de dos maneras:

- Dejar de situarse sobre ella: si no se utiliza, se oculta automáticamente.
- Pulsar el Botón Mute On para silenciar el sonido: al desactivar el sonido, se deshabilita la barra de volumen.

4.4.1.3.3. Barra de reproducción/carga de buffer

Muestra el tiempo que ha transcurrido del vídeo, mediante una barra de color gris, y el porcentaje de bytes recibidos en el buffer, con una barra de color negro. El porcentaje de bytes recibidos nos indica cuánto se ha descargado del vídeo, permitiendo conocer hasta qué parte podemos visualizar. Lógicamente, la barra de reproducción no podrá superar a la barra de carga del buffer.

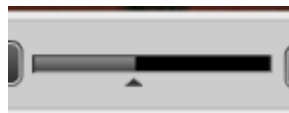


Figura 4.20. Barra de reproducción/carga de buffer

La barra de reproducción también sirve para avanzar y retroceder la reproducción en curso. Desplazando la flecha, se detiene el vídeo y puede seleccionarse el instante de reproducción deseado, lo que también provoca que dejen de aparecer los subtítulos (si se estaban mostrando). Una vez se haya detenido la flecha, la posición de lectura de los subtítulos se modifica para poder leer el subtítulo correcto.

Además, al modificar un subtítulo, se pasa el tiempo de entrada del mismo al reproductor y se sitúa la barra de reproducción en dicho instante. De esta forma, la reproducción del vídeo se colocará en ese instante para mostrar el subtítulo seleccionado y continuar a partir de ese punto.

4.4.1.3.4. Etiqueta de tiempos

Esta etiqueta muestra el instante en el que se está reproduciendo el vídeo y su duración total, utilizando el formato hh:mm:ss, como puede verse en la *Figura 4.21*.

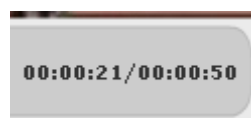


Figura 4.21. Etiqueta de tiempos

Para mostrar el paso del tiempo, se guarda el instante del vídeo en cada fotograma y se transforma al formato de la etiqueta de tiempos, ya que el tiempo que proporciona la transmisión del vídeo está en segundos. Después, se refresca la etiqueta con cada fotograma mostrando el instante correspondiente.

4.4.1.3.5. Botón “Añadir subtítulos”

Al pulsar este botón, se recoge el instante de reproducción del vídeo y se envía al editor de subtítulos. Esto permite utilizar el tiempo exacto de entrada del subtítulo, permitiéndose la creación de subtítulos de una forma más cómoda, sin tener que escribir el tiempo de forma manual.

Como se ha dicho en el apartado 4.4.1.3.4. *Etiqueta de tiempos*, el tiempo que se recoge al pulsar el botón está en segundos pero, al necesitarse el formato hh:mm:ss en el proceso de subtitulado, al pulsar el botón también se realiza esta conversión. Además, antes de enviar el tiempo, se comprueba si el vídeo está detenido. Si no lo está, es necesario pararlo para escribir los demás datos que definen cada subtítulo. Para continuar con la reproducción del vídeo, sólo hay que pulsar el botón Play.

4.4.1.4. Lectura y escritura de subtítulos

Como se ha visto en el apartado 4.4.1.1. *Inicialización*, se comprueba si existe el archivo XML de subtítulos¹¹. Si el archivo ya ha sido creado, el reproductor lee ese archivo y ofrece la posibilidad de mostrar los subtítulos con el vídeo.

Durante el proceso de creación y modificación de los subtítulos, se hará uso de un fichero temporal en el que se irán almacenando las entradas del usuario. Una vez finalizado el proceso, el archivo temporal es utilizado para generar el archivo de subtítulos final.

¹¹ El nombre del archivo XML se toma, al igual que el nombre del vídeo, de la URL en la que se encuentra la aplicación de subtitulado.

Es importante destacar que, al leer el archivo XML, cada subtítulo (tiempos, texto y formato) pasa a tener una posición (primero, segundo...), dato fundamental a la hora de mostrar los subtítulos.

Tras haber leído el archivo temporal, el reproductor debe mostrar los subtítulos si éstos han sido activados. Para ello, en cada fotograma, se comprueba si el instante en el que se encuentra la reproducción del vídeo coincide con los tiempos de entrada y salida del subtítulo.

Si el tiempo de reproducción está entre los dos tiempos, debe aparecer el subtítulo, y si es superior al tiempo de salida, debe desaparecer. Si existen más subtítulos a mostrar, la lectura del XML pasa a la siguiente posición y se hace la misma comprobación para los tiempos de ese subtítulo.

4.4.1.5. Detección del final del vídeo

Al terminar el vídeo, todos los elementos deben volver al estado o posición inicial para poder empezar de nuevo la reproducción de forma correcta. De este modo, al detectarse el final del vídeo, se coloca el tiempo de reproducción al inicio del vídeo y se muestra el Botón Play y habilita la funcionalidad de inicio de reproducción sobre la pantalla de vídeo. También se muestra el Botón Caption Off, ya que los subtítulos comienzan desactivados, y se coloca la lectura del archivo XML al principio del mismo.

4.4.1.6. Diagrama de flujo final

Teniendo en cuenta la etapa de edición de subtítulos, ya que el reproductor estará embebido en dicha aplicación, el diagrama de flujo queda como aparece en la *Figura 4.22*.

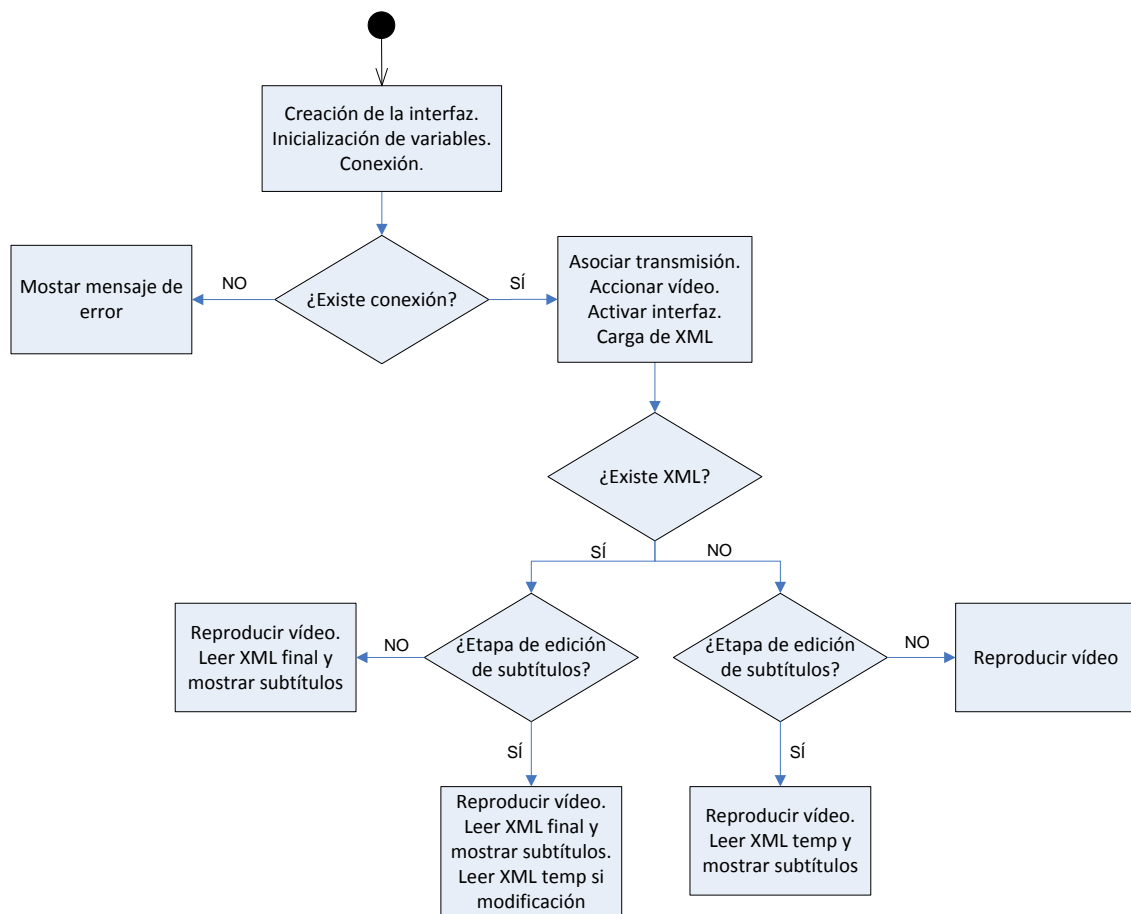


Figura 4.22. Diagrama de flujo del reproductor

4.4.2. Descripción de la aplicación de subtitulado

Para poder acceder a la aplicación desarrollada, como ya se explicó en el apartado 4.2.4. *Integración en el Gestor ARCAMM*, primero se crea un nuevo rol a través de la interfaz de administración de ARCAMM. Una vez creado, se elige ese nuevo rol tras validarse en el Gestor, pasándose a la página principal del rol.

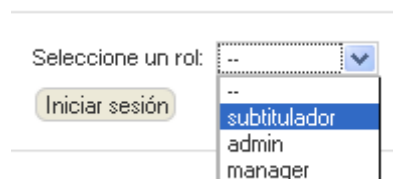


Figura 4.23. Selección del rol “subtitulador”

En la página principal se selecciona un vídeo para pasar a la página ítem y, en ésta, se pincha sobre el icono propio del rol “subtitulador” (Figura 4.24), accediendo a la página principal de la aplicación desarrollada (Figura 4.25).

The screenshot shows the 'audiovisuales' website interface. At the top, there is a header for 'Universidad Carlos III de Madrid' and a date 'Hoy es: Martes, 06 Julio 2010.' Below the header is a navigation bar with links: Inicio, Eventos, Videos/Podcast, Series/Cursos, RSS, ARCA-RSS, and SONIA MARTIN CARMONA. A search bar is present with a 'Buscar' button. The main content area displays a video player with the UC3M logo. To the left of the video player is a sidebar containing metadata for the video, including the title 'Plataformas virtuales 3D para el aprendizaje', the author 'oic@uc3m.es', and the duration '0h. 4m. 37s'. A red box highlights the 'Subtitulador' icon in the sidebar, which is a clapperboard with the word 'SUB' on it. A yellow warning icon and text are also visible above the video player, indicating a migration of content.

Figura 4.24. Página ítem del rol “subtitulador”

Universidad Carlos III de Madrid
Editor de subtítulos

Tiempo inicial (hh:mm:ss.ddd):

Duración (en segundos):

Texto del subtítulo:

Color del texto: Blanco

Añadir subtítulo

Publicar subtítulos: ☐

Tiempo inicial	Texto subtítulo	Duración	Color
----------------	-----------------	----------	-------

(c) Universidad Carlos III de Madrid

Figura 4.25. Página de la aplicación de subtítulos

Como se puede observar en la *Figura 4.25*, para generar un subtítulo el usuario debe rellenar tres campos:

- **Tiempo inicial:** instante de tiempo en el que se quiere que aparezca el subtítulo.
- **Duración:** tiempo que debe seguir mostrándose el subtítulo.
- **Texto del subtítulo:** diálogo o texto que debe mostrar el subtítulo.

Además, el usuario también puede seleccionar el color de letra del subtítulo mediante el menú desplegable **Color del texto**.

Para crear un subtítulo, el subtitulador debe introducir todos los datos y pinchar sobre el icono *Crear subtítulos* (ver *Figura 4.26*), con lo que se guardará el

subtítulo de manera temporal y se visualizará en la tabla que aparece en la parte inferior de la página.



Figura 4.26. Iconos de izquierda a derecha: Crear subtítulos, Crear archivo XML, Borrar archivo XML

Antes de guardar los datos, se comprueba que están correctamente escritos y presentan el formato adecuado. Si no lo están, se muestra un mensaje indicando el error generado. En la *Figura 4.27*, se muestra un ejemplo de un subtítulo, después de crearlo.

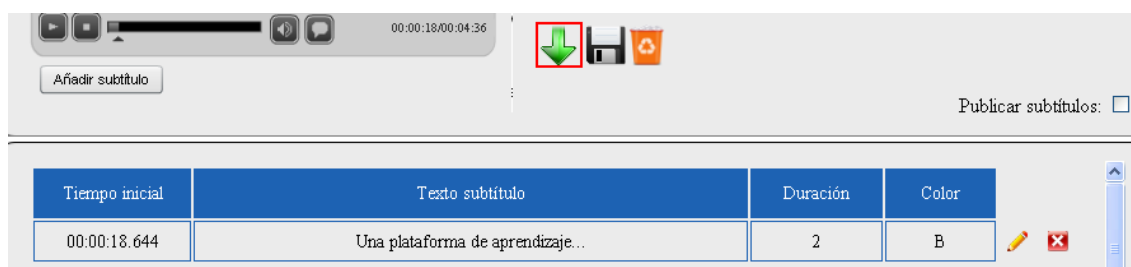


Figura 4.27. Ejemplo de subtítulo

También se comprueba que los subtítulos que se van introduciendo no están superpuestos en el tiempo, es decir, que un subtítulo no empieza ni termina en el intervalo de tiempo de otro subtítulo. Si esto ocurre, la aplicación mostrará un mensaje de error avisando al usuario subtitulador y no creará el subtítulo.

Para que puedan ser corregidos, los datos del subtítulo permanecerán en los campos correspondientes. Una vez rectificados, se procederá a su almacenamiento temporal.

Para obtener una mayor sincronización de los subtítulos con la voz, hay veces que es necesario introducir tiempos con milésimas de segundo. Por esta razón, para ayudar al subtitulador a realizar su trabajo, la aplicación consta de un botón para seleccionar, mientras se reproduce el contenido audiovisual, el tiempo de entrada del

subtítulo. Al pinchar sobre este botón, se detendrá la reproducción y el tiempo aparecerá en el campo **Tiempo inicial**.

Esta funcionalidad, además, evita tener que introducir los tiempos de forma manual.



Figura 4.28. Botón “Añadir subtítulo” y marca de tiempo

Los subtítulos introducidos pueden modificarse o eliminarse utilizando los dos iconos que aparecen al lado de cada uno de ellos en la tabla de visualización (ver *Figura 4.27*). Si se desea editar un subtítulo, tras pinchar sobre el icono correspondiente, los datos aparecerán en los campos a rellenar por el subtitulador, para que éste introduzca los cambios. Para guardar estas modificaciones, hay que pinchar de nuevo en el icono *Crear subtítulos* (*Figura 4.26*).

Al editar el subtítulo, la aplicación sitúa la reproducción del vídeo en el tiempo de entrada del subtítulo a modificar. Si el subtitulador cambia de opinión, puede pinchar en el botón “Cancelar”, que aparece en lugar del botón de editar, para no realizar cambios en el subtítulo.

Los subtítulos que se guardan temporalmente pueden verse en el reproductor de la aplicación, para que el subtitulador pueda visualizar un previo y comprobar que

los subtítulos van a quedar de la manera que él desea. Para ello, el subtitulador debe activarlos haciendo clic sobre el Botón Caption.

Para guardar de manera permanente los subtítulos temporales generados, es necesario crear un archivo de subtítulos. Si no se crea el archivo, los subtítulos desaparecerán al salir de la aplicación. Para crear el archivo, hay que pinchar sobre el icono *Crear archivo XML* (ver Figura 4.26), generándose un archivo XML, cuyo nombre será el mismo que el nombre del vídeo. Este archivo puede crearse en cualquier momento del proceso de edición.

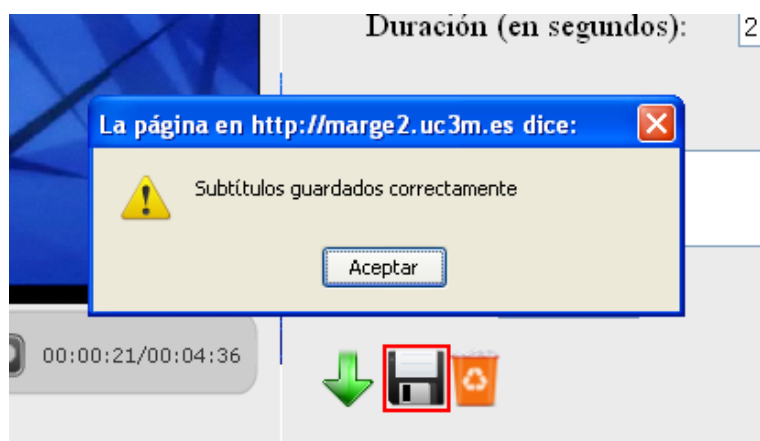


Figura 4.29. Guardar subtítulos en archivo

Si se quiere eliminar el archivo XML existente, basta con pinchar sobre el icono *Borrar archivo XML* (ver Figura 4.26). Este proceso borrará el archivo del servidor, por lo que requiere confirmación por parte del usuario.

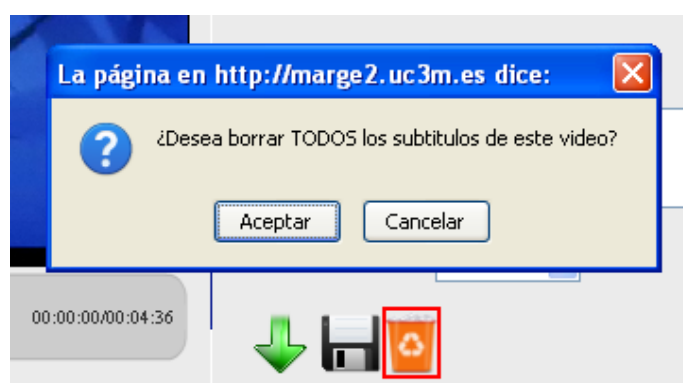


Figura 4.30. Borrar archivo de subtítulos

Si se desea que los subtítulos puedan ser vistos por cualquier usuario mientras se visualiza el contenido audiovisual en la página ítem, el subtitulador debe seleccionar la opción “Publicar subtítulos”. De este modo, el archivo XML se hace público y puede ser visto por cualquier persona que lo desee. Tras seleccionar la opción, hay que hacer clic sobre el segundo icono de la *Figura 4.26* para que la opción quede registrada.

La aplicación desarrollada también permite importar un archivo SubRip generado por otra herramienta de subtitulado. Al pinchar sobre el icono *Importar* (ver *Figura 4.32*), aparece una nueva ventana con un botón “Examinar...”. Al pinchar sobre él, permite elegir el archivo a importar entre el contenido local del equipo.

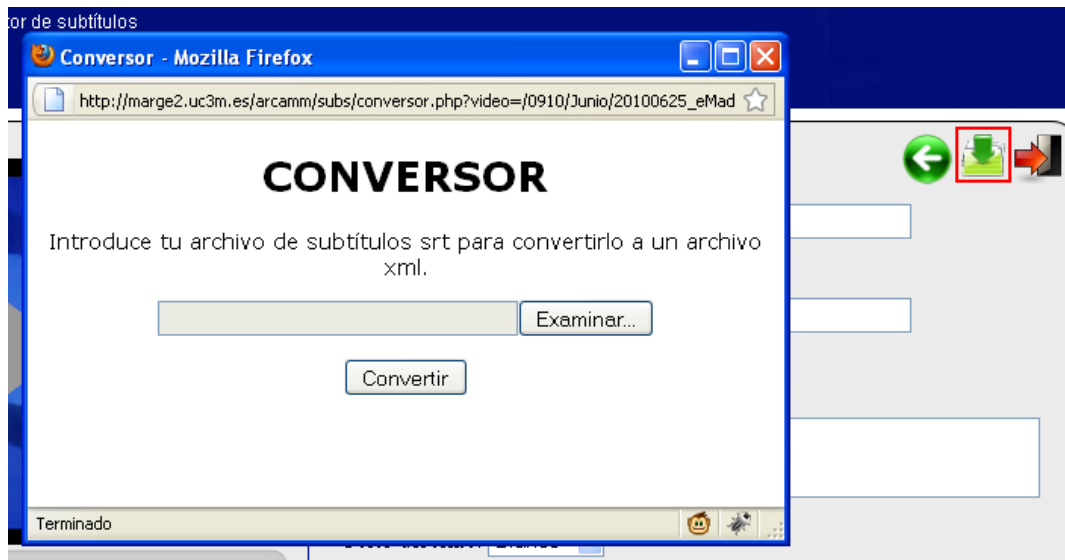


Figura 4.31. Ventana para importar archivo SubRip

Una vez elegido el archivo SubRip, el usuario subtitulador debe pulsar el botón “Convertir” para crear un archivo XML que pueda ser interpretado por el reproductor. Antes de crear el archivo XML, el Conversor verifica que el fichero escogido posee la extensión .srt, que está formado correctamente y que no presenta errores. Si no hay ningún error, se creará el archivo XML, si lo hay, aparecerá un mensaje informando del problema encontrado.



Figura 4.32. Iconos de izquierda a derecha: Volver a página ítem, Importar, Cerrar sesión

La aplicación cuenta también con un icono, *Volver a la página Ítem* (ver Figura 4.32), que permite al subtitulador volver a la página ítem del contenido audiovisual del cual se están generando los subtítulos. De esta forma, el usuario subtitulador puede visualizar el vídeo o presentación con los subtítulos que acaba de crear.

También, si el subtitulador quiere salir de la aplicación y cerrar la sesión iniciada en el portal de ARCMM, puede hacer clic en el icono *Cerrar sesión* (ver Figura 4.32), el cual elimina la sesión y sitúa al usuario en la página principal de ARCMM.

4.5. Resumen del proyecto

En este apartado se muestran el tiempo dedicado a cada una de las fases en las que se ha dividido el desarrollo de este proyecto, mediante un Diagrama de Gantt con la planificación final.

También se incluye una estimación del presupuesto total del proyecto, incluyendo el coste de los recursos, humanos y materiales, empleados.

4.5.1. Planificación final

Al inicio de este proyecto se realizó una planificación de las diferentes tareas a realizar, la cual se ha intentado seguir en todo momento durante la realización de las mismas. Aunque se han producido pequeñas desviaciones, que han demorado la realización de algunas tareas, en líneas generales, se ha cumplido la planificación inicial planteada.

En la *Figura 4.33* se encuentra el Diagrama de Gantt con la planificación final del proyecto.

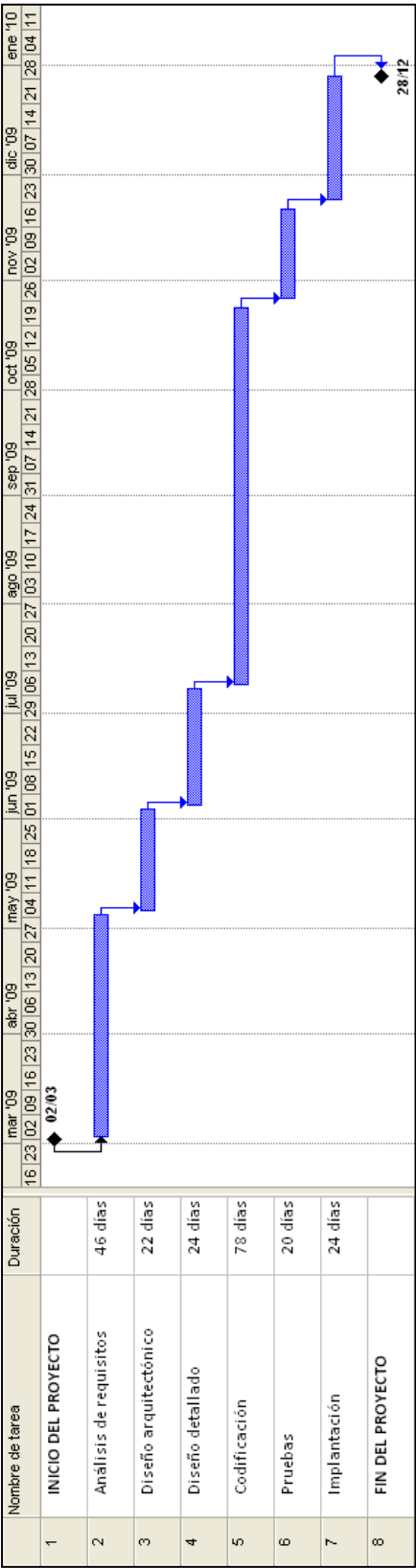


Figura 4.33. Diagrama de Gantt

4.5.2. Resumen de costes

Para estimar el coste de personal, es necesario calcular el número de horas invertidas en el desarrollo total del proyecto. Para ello, se calculan las horas dedicadas a cada una de las tareas, las cuales pueden verse en la *Tabla 4.52*.

Fase	Número de días	Número de horas
Análisis de requisitos	46	184
Diseño del sistema	22	88
Diseño del programa	24	96
Codificación	78	312
Pruebas	20	80
Implantación	24	96
	214	856

Tabla 4.52. Resumen de días y horas por fase

Como se resume en la tabla anterior, el número de horas totales es de 856, obtenido del número de horas trabajadas por cada día (4 horas), sin tener en cuenta los fines de semana y los días festivos.

Cada hora trabajada por un desarrollador de aplicaciones se estima en 12 euros y, por consiguiente, el coste de personal asciende a 10.272 euros.

Para el coste de material no se ha tenido en cuenta el coste del entorno de producción, debido a que ya está desplegado. Por consiguiente, sólo se ha utilizado el coste del material de desarrollo. En la *Tabla 4.53* se muestra el coste de cada material.

CONCEPTO	IMPORTE
Ordenador PC multimedia	800 €
Adobe Flash CS3 Professional	200 €
Contrato de mantenimiento con Adobe Flash	200 €
Coste Total	1.200 €

Tabla 4.53. Coste de material

Con todo esto, el coste total del proyecto toma la cifra de **13.536,96 euros**, como puede verse en la *Tabla 4.54*.

CONCEPTO	IMPORTE
Coste de personal	10.272 €
Coste de material	1.200 €
I.V.A. (16%)	1.835,52 €
Coste Total	13.307,52 €

Tabla 4.54. Coste total del proyecto

4.5.3. Estado del proyecto

En la actualidad, el proyecto está siendo utilizado por personal de la Universidad Carlos III de Madrid. Se subtitulan los vídeos de la cartelera de la Oficina de Información Científica (OIC) y los vídeos de clases semipresenciales.

A día de hoy, se subtitulan alrededor de 10 vídeos a la semana, utilizando las dos formas de subtitulación que fueron diseñadas en la herramienta:

- Generando un archivo de subtítulos desde cero: escribiendo todos los tiempos de entrada y salida y el texto de cada subtítulo.
- Utilizando un archivo de subtítulos SubRip a través de la importación. El archivo SubRip se genera con una herramienta de reconocimiento de voz a la que se le proporciona el audio del vídeo. Si el hablante tiene buena dicción, este reconocedor de voz tiene una tasa de acierto (sin entrenar antes la herramienta) de un 80%.

Utilizando la primera forma de subtitulación, se tarda en subtitular, con un subtitulador entrenado, aproximadamente unos 12 minutos por cada minuto de vídeo. Con la otra forma, se reduce el tiempo, que es de cinco minutos por cada minuto de vídeo, ya que los subtítulos ya están generados y sólo hay que corregir los errores que haya cometido el reconocedor de voz.

CAPÍTULO 5. CONCLUSIONES

Como se ha podido observar con la realización de este trabajo, la accesibilidad de aplicaciones y contenidos en la Web es un área en el que se investiga activamente en la actualidad.

Una de las técnicas para conseguir un contenido audiovisual accesible es el subtitulado. Como ha podido comprobarse, existen múltiples herramientas de subtitulación aunque, por lo general, no poseen todos los requisitos que permiten una subtitulación on-line, con gestión de formato, corrección de errores y que puedan ser utilizadas independientemente del propietario del vídeo a subtitular.

A partir de los objetivos planteados en el apartado 1.2. *Objetivos*, se ha conseguido crear una aplicación capaz de reproducir los vídeos existentes en el Gestor ARCAMM a través de un reproductor, cuya interfaz posee los botones y barras necesarios para una correcta reproducción, esto es, con una interfaz amigable.

Esta aplicación también es capaz de generar un archivo de subtítulos que almacena, en un archivo XML y siguiendo un vocabulario de metadatos definido para su representación (ESUC3M), los subtítulos que introduce el usuario, después de comprobar que no presentan errores.

El reproductor lee este archivo durante la reproducción del vídeo y permite activar/desactivar la visualización de los subtítulos a conveniencia del usuario.

Este archivo puede ser editado accediendo a la aplicación de subtitulado y seleccionando el mismo vídeo. Si existe un archivo para ese vídeo, aparecen los subtítulos en la tabla de visualización, permitiendo realizar las modificaciones pertinentes.

La integración en el Gestor ARCAMM incluye el acceso a través de la validación LDAP, accediendo primero al Gestor y seleccionando el rol “subtitulador”. La aplicación

permite reproducir y editar los subtítulos del contenido multimedia gestionado por ARCAMM, seleccionando el contenido desde la interfaz del Gestor.

Otra función que se ha conseguido es la realización de una interfaz de edición sencilla y fácil de utilizar. La interfaz facilita la edición de subtítulos permitiendo introducir el tiempo de entrada sin necesidad de escribirlo manualmente y realizando un control de superposición. Para el formato del texto se ha incluido la posibilidad de modificar el color de fuente, pudiendo elegir entre cinco colores diferentes.

Por último, proporciona la posibilidad de importar un archivo de subtítulos externo, además de permitir volver a la página ítem del Gestor y cerrar la sesión iniciada como “subtitulador”, aspectos importantes también en la integración con ARCAMM.

A partir de estos resultados se puede concluir que la aplicación desarrollada satisface los objetivos propuestos. Integrada con el Gestor ARCAMM y accesible on-line, la aplicación permite reproducir y subtitular los contenidos compartidos por dicho gestor. La gestión de los subtítulos generados incluye el formateado y el control de errores en el proceso de subtitulación.

Gracias a este trabajo, se ha roto una barrera de accesibilidad que presentaba el sistema de gestión de contenido multimedia de la Universidad Carlos III, permitiendo a usuarios con problemas auditivos o de lenguas extranjeras beneficiarse de los contenidos que ofrece la Universidad.

El cliente, el Área de Audiovisuales del Servicio de Informática de la Universidad Carlos III, ha quedado satisfecho con el resultado de este trabajo. Además, de la utilización del sistema han surgido nuevas ideas por parte de los usuarios, las cuales están recogidas en el siguiente apartado como futuras líneas de desarrollo.

5.1. Futuras líneas de desarrollo

La aplicación desarrollada es susceptible de mejora en varios aspectos resumidos a continuación:

Una de las posibles mejoras consistiría en añadir un módulo de estadísticas que almacene, por ejemplo, el número de usuarios que utilizan la aplicación, el número de vídeos que se subtitulan, qué vídeos se subtitulan más, etc. Este módulo sería interesante para llevar un control del uso de la herramienta.

Para la edición de subtítulos podría añadirse funcionalidad para modificar el formato del texto de los subtítulos, en la que se incluirían distintos tipos de fuente, tamaños y estilos, como negrita, cursiva y subrayado. De esta forma, se le proporcionaría al editor mayor libertad a la hora de diseñar los subtítulos y la posibilidad de adaptarlos a diferentes necesidades.

También sería deseable que la aplicación pudiese importar y exportar el formato *Timed Text Authoring Format 1.0 – DFXP* al ser el formato recomendado por el W3C, cuyas recomendaciones son tomadas como referencia en temas de accesibilidad. Este formato está basado en XML, lo que ayuda en el proceso de edición y lectura del archivo de subtítulos, y es compatible con Flash.

Otra mejora sería la posibilidad de poder incluir más de un archivo de subtítulos para un mismo vídeo o presentación. De este modo sería posible, por ejemplo, subtitular en distintos idiomas o crear archivos con diferentes formatos de subtítulos, mejorando la compatibilidad con otras herramientas existentes.

Finalmente, sería interesante poder automatizar la generación del archivo de subtítulos SubRip, por ejemplo, mediante una herramienta de reconocimiento de voz integrada en el Gestor ARCAMM. De esta forma, se generarían los tiempos de entrada y salida con mayor rapidez y se importaría el archivo, en la aplicación de edición, para

poder corregir los errores producidos por el reconocedor y aplicar el formato deseado. Esto podría implicar en una reducción del tiempo del proceso de subtitulación.

CAPÍTULO 6. BIBLIOGRAFÍA

ADOBE SYSTEMS INCORPORATED. 2007. Programación con ActionScript 3.0. Disponible en http://livedocs.adobe.com/flash/9.0/es/main/flash_as3_programming.pdf, consultado en Mayo de 2010.

ADOBE SYSTEMS INCORPORATED. 2009. *F4V/FLV Technology Center*. Disponible en <http://www.adobe.com/devnet/f4v.html>, consultado en Noviembre de 2010.

ADOBE SYSTEMS INCORPORATED. 2010 a. *Adobe Flash Player*. Disponible en <http://www.adobe.com/es/products/flashplayer/>, consultado en Mayo de 2010.

ADOBE SYSTEMS INCORPORATED. 2010 b. *Adobe Flash Professional CS5*. Disponible en <http://www.adobe.com/es/products/flash/>, consultado en Mayo de 2010.

AE ESTENOTIPISTAS. 2010. Historia - ¿Qué es la estenotipia? Disponible en <http://www.estenotipiaasociacion.com/Copia%20de%20Historia.asp>, consultado en Mayo de 2010.

AEGISUB. 2010. *Aegisub Manual – Highlights*. Disponible en <http://docs.aegisub.org/manual/Highlights>, consultado en Mayo de 2010.

AENOR. 2009. *Accesibilidad Universal*. Disponible en http://www.aenor.es/documentos/certificacion/folletos/w_508_ACCESIBILIDAD_UNIVERSAL.pdf, consultado en Mayo de 2010.

AENOR. 2003. Norma UNE 153010:2003: Subtitulado para personas sordas y personas con discapacidad auditiva. Subtitulado a través del teletexto. Disponible en <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0029761&PDF=Si>, consultado en Mayo de 2010.

AENOR. 2005. Norma UNE 153020:2005: Audiodescripción para personas con discapacidad visual. Requisitos para la audiodescripción y elaboración de audioguías. Disponible en <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0032787&PDF=Si>, consultado en Mayo de 2010.

AENOR. 2007. Norma UNE 139804:2007: Requisitos para el uso de la Lengua de Signos Española en redes informáticas. Disponible en <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0040404&PDF=Si>, consultado en Mayo de 2010.

AGREGA. 2009. Agrega – Acerca de Agrega. Disponible en http://www.proyectoagrega.es/default/acerca_de.php, consultado en Junio de 2010.

ÁLVAREZ, MIGUEL ÁNGEL. 2004. Introducción al manual del lenguaje PHP en su versión 5. Disponible en <http://www.desarrolloweb.com/articulos/1696.php>, consultado en Mayo de 2010.

ANDRÉS, MARTA. 2009. Gestión de Campeonatos de Competición Interna. Disponible en <http://e-archivo.uc3m.es/handle/10016/6952>, consultado en Mayo de 2010.

APACHE. 2009. *Apache HTTP Server Project*. Disponible en <http://httpd.apache.org/>, consultado en Mayo de 2010.

APPLE. 2010. *iTunes U – What is iTunes U*. Disponible en <http://www.apple.com/education/itunes-u/what-is.html>, consultado en Junio de 2010.

BOOCH, G. 2006. UML: El lenguaje unificado de modelado: guía del usuario. Editorial: Addison-Wesley. ISBN: 9788478290765.

CARRERAS, OLGA. 2008. WCAG 2.0. Disponible en <http://olgacarreras.blogspot.com/2007/02/wcag-20.html>, consultado en Mayo de 2010.

CRUZ, FRANCISCO & IBÁÑEZ, NICOLÁS. ARCA: Federación de contenidos multimedia. Disponible en <http://www.mundointernet.es/IMG/pdf/ponencia72.pdf>, consultado en Junio de 2010.

DIAZ, JORGE. 2001. La traducción audiovisual: el subtitulado. Editorial: Ediciones Almar. ISBN: 9788474550696.

DRUPAL. 2010. *Drupal Home Page*. Disponible en <http://drupal.org/>, consultado en Junio de 2010.

DSPACE. 2008. *DSpace – Why Use DSpace*. Disponible en <http://www.dspace.org/why-use-dspace/why-use/>, consultado en Junio de 2010.

EDITPLUS. 2010. *EditPlus Home Page*. Disponible en <http://www.editplus.com>, consultado en Mayo de 2010.

EGUÍLUZ, JAVIER. 2009. Introducción a JavaScript. Capítulo 1. Introducción. - ¿Qué es JavaScript? Disponible en <http://www.librosweb.es/javascript/>, consultado en Mayo de 2010.

FERNÁNDEZ-TOSTADO, TATIANA. 2010. Estudio tecnológico y diseño arquitectónico de un Sistema de Gestión de Esquemas Semánticos basados en Ontologías. Disponible en <http://e-archivo.uc3m.es/handle/10016/6660>, consultado en Mayo de 2010.

GENNIK, JONATHAN. 2004. *SQL: Pocket Guide*. Editorial: O'Reilly & Associates. ISBN: 9780596005122.

GNU OPERATING SYSTEM. 2010. *GNU General Public License, version 2*. Disponible en <http://www.gnu.org/licenses/gpl-2.0.html>, consultado en Junio de 2010.

HERRERA, ROLANDO. 2007. PHP5 VS PHP4 en Revista de Software Libre de la UCI. Disponible en <http://revistauxi.files.wordpress.com/2007/10/uxi7.pdf>, consultado en Mayo de 2010.

HISOFTWARE. 2009. Hi-Caption Studio. Disponible en http://www.hisoftware.com/products/pdfs/Hi-Caption_datasheet_3-09.pdf, consultado en Mayo de 2010.

HISOFTWARE. 2010. *HiSoftware Home Page*. Disponible en <http://www.hisoftware.com/>, consultado en Mayo de 2010.

INTECO. 2004. Norma UNE 139803:2004: Aplicaciones informáticas para personas con discapacidad. Requisitos de accesibilidad para contenidos en la Web. Disponible en http://www.inteco.es/Accesibilidad/Normativa_1/Descarga/DescargaUNE_139803, consultado en Mayo de 2010.

INTERNATIONAL TELECOMMUNICATION UNION. 2005. ITU H.264: *The Advanced Video Video Coding Standard*. Disponible en <http://www.itu.int/osg/spu/newslog/ITU+H264+The+Advanced+Video+Coding+Standard.aspx>, consultado en Noviembre de 2010.

LAMARCA, MARÍA JESÚS. 2009. Hipermedia/Multimedia – Hipertexto, el nuevo concepto de documento en la cultura de la imagen. Disponible en <http://www.hipertexto.info/documentos/hipermedia.htm>, consultado en Junio de 2010.

MATROSKA. 2010 a. *SRT Subtitles*. Disponible en <http://www.matroska.org/technical/specs/subtitles/srt.html>, consultado en Noviembre de 2010.

MATROSKA. 2010 b. *SSA/ASS Subtitles*. Disponible en <http://www.matroska.org/technical/specs/subtitles/ssa.html>, consultado en Noviembre de 2010.

MEDIAMOSA. 2009. *MediaMosa. An Introducion*. Disponible en http://www.mediamosa.org/sites/default/files/Mediamosa_an_introduction.pdf, consultado en Junio de 2010.

MICROSOFT CORPORATION. 2003. *Understanding SAMI 1.0*. Disponible en <http://msdn.microsoft.com/en-us/library/ms971327.aspx>, consultado en Mayo de 2010.

MICROSOFT CORPORATION. 2010. *Windows Media codec*. Disponible en <http://msdn.microsoft.com/es-ES/library/ff819508%28v=VS.85%29.aspx>, consultado en Noviembre de 2010.

MORENO, LOURDES & RUIZ, BELÉN & MARTÍNEZ, PALOMA & CARRERO, JUAN MANUEL & MARTÍNEZ, JUAN RAMÓN. 2008. Accesibilidad a los contenidos audiovisuales en la Web. Una panorámica sobre legislación, tecnologías y estándares. Disponible en http://www.cesya.es/files/documentos/accesibilidad_contenidos.pdf, consultado en Mayo de 2010.

NATIONAL CENTER FOR ACCESIBLE MEDIA. 2009. *MAGpie – MAGpie2 Help Contents*. Disponible en http://ncam.wgbh.org/invent_build/web_multimedia/tools-guidelines/magpie2helpcontents, consultado en Mayo de 2010.

OBJECT MANAGEMENT GROUP. 2010. *OMG Home Page*. Disponible en <http://www.omg.org>, consultado en Noviembre de 2010.

OpenLDAP. 2010. *What is LDAP?* Disponible en <http://www.openldap.org/faq/data/cache/29.html>, consultado en Octubre de 2010.

OVERSTREAM. 2009. *Overstream Help*. Disponible en <http://www.overstream.net/help.php>, consultado en Mayo de 2010.

PÉREZ, ALEJANDRO ALFONSO. 2007. Desarrollo de herramientas web de gestión docente. Disponible en <http://repositorio.bib.upct.es/dspace/bitstream/10317/179/1/pfc2475.pdf>, consultado en Mayo de 2010.

PHP. 2010. php - ¿Qué es PHP? Disponible en <http://www.php.net/manual/es/intro-whatis.php>, consultado en Mayo de 2010.

PUMUKIT. 2009. PuMuKIT *Home Page*. Disponible en <http://www.pumukit.uvigo.es>, consultado en Junio de 2010.

REINO, ALFREDO. 2007. Manual de XML. Disponible en <http://www.desarrolloweb.com/manuales/manual-introduccion-xml.html>, consultado en Mayo de 2010.

RODRÍGUEZ-GAIRÍN, JOSEP-MANUEL & SULÉ, ANDREU. 2008. DSpace: un manual específico para gestores de la información y la documentación. Disponible en <http://www.ub.es/bid/20rodri2.htm>, consultado en Junio de 2010.

STEEPLE. 2010. Steeple *Home Page*. Disponible en <http://steeple.oucs.ox.ac.uk/>, consultado en Junio de 2010.

SPINK, AMANDA & JANSEN, BERNARD J. 2006. *Searching multimedia federated content web collections*. Disponible en <http://www.emeraldinsight.com/journals.htm?articleid=1576307&show=html>, consultado en Junio de 2010.

TIENDA APPLE EMAGISTER.COM. 2009. iTunes U llega a España. Disponible en <http://apple.emagister.com/itunes-u-llega-a-espana/>, consultado en Junio de 2010.

UNIFIED MODELING LANGUAGE. 2010. *UML Resource Page*. Disponible en <http://www.uml.org>, consultado en Mayo de 2010.

UNIVERSIDAD CARLOS III. 2009 a. IPTV UC3M. Disponible en <http://www.uc3m.es/portal/page/portal/informatica/IPTV-UC3M>, consultado en Junio de 2010.

UNIVERSIDAD CARLOS III. 2009 b. ProyectoARCA – Documentación. Disponible en <https://marge2.uc3m.es/arca/doc.php?mod=doc>, consultado en Junio de 2010.

URUWORKS. 2010. Subtitle Workshop. Disponible en <http://www.urusoft.net/products.php?cat=sw&lang=2>, consultado en Mayo de 2010.

VOCES, RAMÓN. 2008. El contenido audiovisual: otro reto para la accesibilidad web. Disponible en <http://www.ub.es/bid/21/voces2.htm>, consultado en Mayo de 2010.

WGBH, MEDIA ACCESS GROUP. 2010. Subtitulación y descripción en Internet. Disponible en http://main.wgbh.org/wgbh/pages/mag/spanish/resources/guides/mag_guide_vol13.html, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2002. *XML Pointer Language (XPointer)*. Disponible en <http://www.w3.org/TR/xptr/>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2005. *Web Accessibility initiative* – Introducción a la Accesibilidad Web. Disponible en <http://www.w3c.es/traducciones/es/wai/intro/accessibility>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2006 a. *Extensible Markup Language (XML) 1.1 (Second Edition) - 2.8 Prolog and Document Type Declaration*. Disponible en <http://www.w3.org/TR/2006/REC-xml11-20060816/#sec-prolog-dtd>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2006 b. *Extensible Stylesheet Language (XSL) Version 1.1*. Disponible en <http://www.w3.org/TR/xsl/>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2006 c. *Web Accessibility initiative* – Componentes esenciales de Accesibilidad Web. Disponible en <http://www.w3c.es/Traducciones/es/WAI/intro/components>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2007 a. *XML Path Language (XPath) 2.0*. Disponible en <http://www.w3.org/TR/xpath20/>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2007 b. *XML Query (XQuery) Requirements*. Disponible en <http://www.w3.org/TR/xquery-requirements/>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 a. El W3C de la A a la Z. Disponible en <http://www.w3c.es/divulgacion/a-z/>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 b. Guía Breve de Accesibilidad Web. Disponible en <http://www.w3c.es/divulgacion/guiasbreves/accesibilidad>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 c. Guía Breve de Tecnologías XML. Disponible en <http://www.w3c.es/divulgación/guiasbreves/tecnologiasxml>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 d. Guía Breve de XHTML. Disponible en <http://www.w3c.es/divulgacion/guiasbreves/XHTML>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 e. *Synchronized Multimedia*. Disponible en <http://www.w3.org/AudioVideo>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2008 f. *Synchronized Multimedia Integration Language* (SMIL 3.0). Disponible en <http://www.w3.org/TR/SMIL/>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 a. *Extensible Markup Language* (XML). Disponible en <http://www.w3.org/XML/>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 b. *Scalable Vector Graphics* (SVG). Disponible en <http://www.w3.org/Graphics/SVG>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 c. *Synchronized Multimedia Activity Statement*. Disponible en <http://www.w3.org/AudioVideo/Activity.html>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 d. *Timed Text Working Group*. Disponible en <http://www.w3.org/AudioVideo/TT/>, consultado en Mayo de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 e. *XML Linking Language (XLink) Version 1.1*. Disponible en <http://www.w3.org/TR/xlink11/>, consultado en Octubre de 2010.

WORLD WIDE WEB CONSORTIUM. 2010 f. *XML Schema*. Disponible en <http://www.w3.org/XML/Schema>, consultado en Octubre de 2010.

CAPÍTULO 7. GLOSARIO

EMISIÓN EN DIRECTO

Distribución de audio y/o vídeo de un evento que está siendo emitido en el mismo momento en el que se reproduce. Este evento se visualiza a través de un reproductor incrustado en una página web.

FLUJOS RSS

Conjunto de archivos RSS que se envía por Internet. RSS es un formato de datos que se utiliza para enviar información actualizada a los suscriptores de un sitio en Internet.

LECTOR DE PANTALLA

Software que intenta interpretar lo que se muestra en una pantalla. Posteriormente, la interpretación se presenta al usuario mediante un sintetizador de voz, iconos sonoros o una salida en braille.

MEDIATECA

Espacio donde se tienen, organizan y ponen a disposición del público materiales multimedia y medios de comunicación social. También se encarga de preservar adecuadamente, para su uso, dicho material.

PODCAST/VODCAST

Distribución a petición de audio/video para, generalmente, dispositivos portátiles. El archivo se descarga y se almacena en un dispositivo, portátil o no, para reproducirlo posteriormente.

STREAMING

Distribución de audio y/o vídeo por Internet de manera que, mientras se va descargando dicho contenido, se puede ir reproduciendo. Para ello, en el cliente existe

un buffer donde se va almacenando el contenido y el cual va enviando al reproductor la información cuando considera que hay una cantidad suficiente de ella.

TECNOLOGÍAS ASISTIVAS

Aquellos productos, equipamientos o sistemas técnicos diseñados y desarrollados especialmente para personas con discapacidad, cuyo objetivo es mejorar su autonomía personal y su calidad de vida.

VÍDEO BAJO DEMANDA (en inglés Video on Demand: VoD)

Modalidad de distribución de vídeo que se basa en la libertad de elección, por parte del usuario, del vídeo que se desea ver en un momento determinado. De esta forma, la programación seleccionada se distribuye de manera personalizada.

Dependiendo del sitio que ofrezca el VoD, el vídeo podrá almacenarse o no.

WebTV e IPTV

WebTV es la integración de vídeo en páginas web y portales de Internet utilizando tecnologías de televisión por Internet que permiten la distribución del vídeo de manera on-line, utilizando el protocolo de Internet IP.

IPTV es, según la UIT (Unión Internacional de Telecomunicaciones), un conjunto de servicios multimedia (televisión, vídeo, audio, texto, gráficos y datos) que son distribuidos por una red IP.

La diferencia principal con WebTV es que en IPTV no es necesario usar la Web para visualizar los contenidos: éstos pueden verse en un televisor, por ejemplo. En IPTV, IP hace referencia a un método de envío de información a través de una red segura y fuertemente administrada.

Sin embargo, en la actualidad, pocas personas conocen bien las diferencias entre estos dos términos, incluso personas que trabajan con estas tecnologías. Por eso, suele pensarse que IPTV y WebTV son lo mismo.